University of Cincinnati

Date: 6/29/2015

I, Kasun M Samarasinghe , hereby submit this original work as part of the requirements for the degree of Doctor of Philosophy in Electrical Engineering.

It is entitled:

Sparse Signal Reconstruction Modeling for MEG Source Localization Using Non-convex Regularizers

Student's name: **Kasun M Samarasinghe**

This work and its defense approved by:

Committee chair: H. Howard Fan, Ph.D.

Committee member: Donald French, Ph.D.

Committee member: William Wee, Ph.D.

Committee member: Jing Xiang, Ph.D.

Committee member: Xuefu Zhou, Ph.D.

16224

Sparse Signal Reconstruction Modeling for MEG Source Localization using Non-convex Regularizers

A dissertation submitted to the

Division of Research and Advanced Students
of the University of Cincinnati

in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in the Department of Electrical Engineering and Computing Systems
of the College of Engineering & Applied Science
University of Cincinnati
June 2015

by

Kasun Maduranga Samarasinghe

B.S. Electrical Engineering
University of Peradeniya, Peradeniya, Sri Lanka
April 2008

Thesis Advisor and Committee Chair: **Prof. Howard H. Fan**

ABSTRACT

This thesis introduces the usage of non-convex based regularizers to solve the underdetermined MEG inverse problem. The signal to be reconstructed is considered to have a structure which entails group-wise sparsity and within group sparsity among its covariates. We discuss the usage of l_2 norm regularization and smoothed l_0 (SL0) norm regularization to impose group-wise and within group sparsity respectively. In addition, we introduce a novel criterion, which if satisfied, guarantees global optimality while solving this non-convex optimization problem. We use proximal gradient descent as the method of optimization as it promises faster convergence rates. Initially, we show that our algorithm successfully recovers sparse signals with a smaller number of measurements than the conventional l_1 regularization framework. We also support this claim using MEG source localization simulations and extend the reconstruction for both stationary and non-stationary signals.

Next, we formulate a global convergence analysis for the novel algorithm. Finally, we incorporate novel information criteria techniques and concepts of duality to find the best set of regularization parameters and a proper stopping criterion respectively. We were able to successfully illustrate that the regularization parameters (models) with lower information criteria performs better than the ones with higher information criteria. Also, concepts of duality provides the necessary tools to determine when to stop the algorithm, which is an important contribution considering the non- differentiability of the objective function.

ACKNOWLEDGMENTS

First and foremost I would like to thank my mentor Dr. Howard Fan for his leadership and guidance over the past four years. His encouragement, support, feedback and suggestions have helped me immensely to better myself as a researcher while also becoming a more successful PhD student. I would also like to thank Dr. Jing Xiang for his support and kind advice throughout my PhD process. I am greatly thankful to him for letting me to access the CCHMC resources and helping me to learn the basics of MEG. Furthermore, I would like to thank Dr. Donald French for his valuable input and advice throughout my PhD work. I would also like to thank Dr. William Wee and Dr. Xuefu Zhou for taking the time off from their busy schedules to participate on my committee and their valuable inputs.

In addition, I would like to thank Joe and Tim for making my workplace enjoyable and allowing me a more flexible schedule. I would also like to thank my colleague Amal for the valuable discussions we had related to Compressive Sensing and also for being a great friend throughout my PhD years. Special thanks to Arielle and Tom for helping me through my PhD career and being there for me at certain difficult times.

Most importantly I would like to thank my parents, my brother and sister, and my uncle and aunt for their continuous encouragement and support for my higher education.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 Introduction to E/MEG – Brain Source Localization

Magneto-encephalography (MEG) and Electro-encephalography (EEG) are two of the most commonly used non-invasive techniques to solve the “inverse problem” of brain source localization from MEG measurement. MEG and EEG observe the magnetic and electric fields near the scalp surface generated by the neuronal sources inside the brain, respectively. Locating such neuron sources can aid MDs in diagnose and treatment of certain neurological diseases. The advent of these techniques thus has helped profoundly to analyze and prevent brain related diseases in clinical environments in non-invasive ways. For example, epileptic patients suffer from recurrent seizures that occur at unpredictable times without any warning. These seizures are transient anomalies in the brain’s electrical activity. M/EEG can be used to early detect and localize these epileptic loci, and therefore measures can be taken to help patients to reduce the risk of sustaining physical injuries and potential result of death. Some of the other useful applications of using M/EEG are diagnosing brain tumors, detecting abnormal brain states or to classify sleep stages and understanding the functionality and the brain responses related to languages, emotions and vision.

A typical neuron, as shown in Figure 1.1, consists of three main parts: the cell body (soma), dendrites and axon. Even though the cell body can have numerous dendrites, it will only give rise to one axon. “Synaptic signaling” is a structure in the nervous system that allows a neuron to pass an electrical or chemical signal to another cell. These synaptic signals are received by the cell body and the dendrites, and then transmitted to a neighboring cell using the axon. Therefore, a typical synapse can be referred to as a contact between the axon of one neuron and the cell body/dendrites of another. The latter is called the post-synaptic neuron. If this synapse received at the post-synaptic neuron is large enough during a short period of time, the neuron will generate an electric pulse called an action potential. These synapses can be either excitatory (the post-synaptic neuron is more likely to fire an action potential) or inhibitory (the post-synaptic neuron is less likely to fire an action potential).

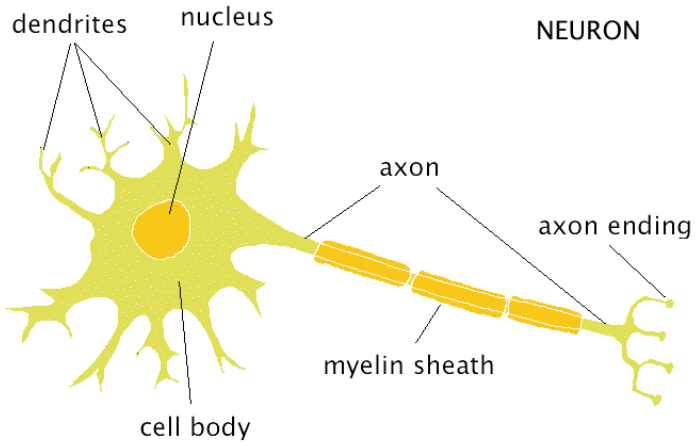


Figure 1.1: Structure of a typical Neuron [72]

Usually, the apical dendrites in cortical pyramidal cells of the brain cortex are assumed to generate the strongest signals [73]. However, one dendrite is far too weak to produce a measurable signal. Therefore, for modeling purposes, a synchronous collection of thousands of dendrites are collectively gathered as one measurement - “a dipole”. These dipoles will act as current generators, and according to the Maxwell’s equations, they would create an electric field and an orthogonal magnetic field which will be captured by EEG and MEG sensors respectively with high temporal resolution.

Unfortunately, even though the number of active source locations could be only a few, the number of potential source locations (positions for a potential current dipole) in the entire area of interest can be relatively high. In MEG modeling, the number of sensors is usually around two hundred, which is much less than the total number of potential dipole locations in the brain. Therefore, this inverse problem of finding the position, amplitude and the orientation of the unknown and active current dipoles becomes underdetermined (we will explain more on the orientation properties of dipoles in Chapter 5). The current dipoles are usually assumed to be positioned on the cortex of the brain, and these positions are represented by vertices in the cortical mesh [1].

1.2 Previous Work

Existing approaches to solving this under-determined system can be categorized into three classes: Dipole Fitting methods [2, 3, 4], Scanning methods [5, 6] and Imaging

methods. Even though Dipole Fitting is the most commonly used technique in the clinical setting, it requires knowledge of the number of active dipoles in advance [7]. Examples of Scanning methods include Beamforming methods [6], Mutiple Signal Classification method (MUSIC) [5], and Maximum Likelihood Estimation (MLE) methods etc… Beamforming methods do not require this assumption; as a result, they give a more impartial estimation. However, Beamforming methods are unable to distinguish between two correlated source activations. Usually, there are two types of correlations that can interfere with the beamforming estimation process: the correlation between different dipoles and the correlation within a dipole component. The latter correlation is caused by continuous rotation or wobbling of the dipole during the measurement. Due to this drawback, Imaging methods were introduced as an alternative.

1.2.1 Imaging methods – Advantages and Drawbacks

Imaging methods assume that primary sources can be represented as linear combinations of neuron (dipole) activities. Also, for a given task within a given time period, it assumes that only a few dipoles are active at the same time. Examples of Imaging methods include Minimum Norm Estimate (MNE) [8, 1, 9], LORETA/ sLORETA [10], Minimum Current Estimate (MCE) [11], FOCUSS with the use of Iterative Weighted Lease Squares algorithm (IRLS) [13] etc. All these methods include a norm minimization procedure, which is referred to as a norm-prior. Imaging methods are not affected by the correlation property of the sources, neither it requires the knowledge of the number

of active dipoles. Therefore imaging methods are the most practical methods, providing the best solutions so far. From now on we will concentrate on imaging methods.

The MNE method uses l_2 norm as its norm-prior, which makes the minimization model convex and differentiable. This makes the estimation extremely fast. However, l_2 norm based methods have numerous limitations. They fail to recognize focal activities and tend to smear the active dipole positions. For example, MNE results are often too diffused for applications such as early detection of epileptic foci. Also, they tend to misplace deeper sources onto the outermost cortex, causing increased ambiguity.

LORETA uses the regularization concept of ridge-regression, or otherwise referred to as a special case of Elastic-net approach. This itself is similar to MNE such that the overall cost function is convex, and hence a global optimum can be found quickly. However, just like the MNE approach, the results are much more diffused and are not suitable for focal-source localization.

Due to these drawbacks, researchers moved their focus more on methods that would encourage spatially-sparse behavior, using the assumption that only a few source locations are activated at a given time. As a result, MCE was introduced, which is also referred to as LASSO (Least Absolute Shrinkage and Selection Operator) [13]. MCE/ LASSO uses l_1 as the norm-prior, inducing sparsity on the solution. Around the same

time Gorodnitsky et al. [12] introduced FOCUSS, which uses the l_p norm-prior $(0 < p < 1)$.

These regularization methods can be expressed by the following inverse problem and its corresponding solution as:

$$Y = AX + E$$

$$argmin_X \left[\left\| Y - AX \right\|_F^2 + \lambda \Omega(X) \right]$$

where $Y \in \mathbb{R}^{M \times T}$ is the sensor measurement matrix, $X \in \mathbb{R}^{N \times T}$ is the source matrix to be solved, $A \in \mathbb{R}^{M \times N}$ is the Lead-field/ Gain matrix, N is the number of sources, M is the number of sensors (with $N >> M$), T is the number of time samples, $E \in \mathbb{R}^{M \times T}$ is white Gaussian noise, λ is the Regularization parameter and Ω is the regularizer/ penalty function. Also, the Frobenius norm of X is defined as $\|X\|_F = \sqrt{tr\left(X^T X\right)}$. An in-depth explanation about the definitions and the behaviors of norm functions are provided in Chapter 2.

The Imaging methods will differ from each other on how they define the penalty function Ω. For MNE and LORETA the penalty function can be generally defined as $\Omega = \|WX\|_F^2$, where W is the weighting function . For MNE W would be an identity matrix, and for LORETA W would be the discrete spatial Laplacian operator. For MCE/ LASSO and FOCUSS the penalty function would be defined as $\Omega = |X|_p$ for $p = 1$ and

$0 < p < 1$ respectively. As will be shown later in Chapters 2, using an l_p norm, $p < 1$, promotes sparsity in the solution.

However, MCE and FOCUSS methods promote sparsity at each time sample - penalizing both spatially, and temporally. This may cause a failure in recovering the exact time-courses of cortical sources since the time course of a cortical source may not be sparse. Hence, in contrast to MNE, MCE will result in "spiky" discontinuities.

In the literature of Source Localization, it is collectively agreed that the neural activations are spatially focal (sparse) and temporally smooth (not sparse) [14]. If we look at the aforementioned methods, they tend to focus only on one aspect, sacrificing the quality of the other. As a result, a great deal of research has been devoted to finding novel spatio-temporal regularization methods, which would encourage sparsity while preserving the temporal smoothness.

1.2.2 Mixed Norms

As a solution to the above problem, mixed norm sparsity-inducing priors were introduced. Haufe et al. [15] introduced a method, which promotes spatial sparsity via l_1 norm-prior, while l_2 norm-prior is used for the orientations. Similarly, Ou et al. [16], used l_1 norm-prior for spatial sparsity while l_2 norm-prior was used on both orientations and time samples. They also used SVD to compact the signal subspace – significantly reducing the number of time samples. Therefore it is apparent that the

underlying reason for the success of these solvers is their adaptation to the structure of the sparsity of the problem. Similar methods were discussed by Friston et al. [17], Aurannen et al. [18] and Jeffs et al. [19], where the latter used various l_p norm-priors with $0 < p < 1$ and $1 < p < 2$.

Despite their usefulness, most of the aforementioned solvers were computationally slow in finding the estimates; hence a growing interest for faster minimization algorithms arose. Novel minimization techniques for non-differentiable cost models like Proximal Gradient Methods [65, 66, 20] outperform conventional Second Order Cone Programming (SOCP) reduction techniques and interior-point methods in computational speed.

Pioneering work in this regard was done by Gramfort et al. [20, 21], where they introduced various types of mixed norms depending on the structure of the inverse problem. They also introduced faster Proximal Splitting Methods during the optimization process saving significant amount of computational time. Among those mixed norms, two-level sparsity-inducing priors were used for both spatial and time domains, while three-level sparsity-inducing priors were used for spatial, time and "experimental conditions" domains. In their experiments, they considered a somatosensory data-set, where the stimulus was delivered as a square-waved electrical pulse on each finger [20]. Each finger stimulated was considered an experimental condition. They demonstrated that by using the mixed norm approach, they were able

to reconstruct similar size active sources for each finger in the somatosensory cortex, while the l_2 norm priors gave different size active sources for each finger. Therefore, they were able to successfully show the importance of using mixed norm priors as opposed to individual priors.

1.3 Structured Sparsity in M/EEG Source Localization

1.3.1 Non-Stationary behavior of Neuronal brain sources

Most of the above methods, which obtain temporal smoothness via l_2 norm-prior, rely heavily on the assumption that the source activation remains the same through-out the time interval of interest [16, 22, 20]. For example, Ou et al. [16] obtains temporal smoothness through temporal basis functions using SVD. The validity of this step strongly relies on the assumption that sources are stationary.

Even though this assumption is valid for small time windows, in a realistic setting multiple sources will be switching “ON” and “OFF” during the time window of interest. Pioneering work combining sparsity-inducing methods and non-stationary focal source localization was done by Gramfort et al. [21], where they enforced sparsity on time-frequency decompositions of the sources. They used the assumption that each active dipole is a linear combination of a limited number of Gabor atoms. Since a Gabor atom is localized in time, user can now define the time window of interest depending on the

functional behavior of the source. However, this algorithm needs to compute Gabor transforms at every iteration, causing a high computational complexity.

1.3.2 Structure based on Regions of Interest (ROI's)

Another issue that needs to be considered is ROI. In order to understand the functional properties of the brain, it is important to be able to distinguish brain source activation regions that depend on different tasks, i.e., there is a need to distinguish ROI's for different tasks. Unfortunately, conventional inverse solvers fail to distinguish ROI's that are in close proximity. For example, ROI's for the visual system with retinotopic mapping can be determined with fMRI [23,24]. These regions correspond to distinct visual areas on the brain depending on the visionary function. However, due to their close proximity, during M/EEG inversion some regions might be subjected to aliasing, which increases ambiguity.

1.3.3 Group Sparsity concept for M/EEG Source Localization

Group sparsity concept was initially introduced by Yuan et al. [25] and it was referred to as Group Lasso. While Lasso was able to zero-out single sources, Group Lasso could force groups of sources to zero. Hence, Group Lasso was highly advantageous when the structure of the problem can be modeled as groups. In other words, Group Lasso uses the *a-priori* structure information of the problem at hand to improve the quality of the estimation. Therefore, for non-stationary source activations and estimating ROI's, Group Lasso naturally provides a better solution as compared to the conventional mixed

norm solutions. Also, it is observed [26] that when we have measurements from different subjects (can be the same patient at a different time of measurement or different patients), setting an *a-priori* Group structure helps to settle disputes about the functional behavior of brain regions.

In addition, on a general basis, provided that the group structure is correctly guessed, Group Lasso is more robust with stochastic noise as compared to standard Lasso. Furthermore, it is proven [27] that Group Lasso requires a smaller sample size to satisfy the sparse eigen-value condition required in sparsity analysis compared to standard Lasso. A concise analysis of some of these claims is provided in Chapter 2, while a detailed description can be found in [27].

It is important to note that Group Lasso shows superior reconstruction only when the group structure is correctly guessed *a-priori*. In [27] Huang et al. demonstrated that when the group structure is guessed incorrectly, Group Lasso showed inferior results to Lasso.

In the M/EEG context, this *a-priori* structure can be obtained by using functional MRI (fMRI) measurements [26]. For example, using fMRI [74, 75], topographic maps can be obtained for the visual system, which correspond to distinct visual areas that have different functionalities. This *a-priori* structure can then be used for Group Lasso reconstruction.

1.4 Our Contribution

This phenomenon of structure among brain source activities propelled us to introduce group based sparsity into the M/EEG source localization paradigm. Our contribution in this thesis is three-fold.

Firstly, we introduce the Group sparsity concept and its extensions as we believe it's a relatively novel idea in the M/EEG Source Localization framework. While this work was carried out, we came across similar approaches by Jair Monotoya et al. [28] and the PhD thesis work by Michale Kim [26]. However, we focus on more challenging non-convex l_0 - norm approximation based priors as opposed to the conventional and less robust l_1 - norm-priors. To the best of our knowledge, Gaussian based l_0 norm approximations have not been used for the M/EEG inverse problem in previous work. In Chapter 2 and 3 we provide a theoretical analysis of why "l_0 - norm" is preferred to its counterpart l_1 norm methods.

Secondly, as a part of our main contribution, we provide a thorough statistical analysis of the algorithm we have introduced. Furthermore, we introduce the concepts of Group Sparsity and Sparse Group Sparsity to the cost minimization model, making it more robust for a wide-range of underdetermined problems. Within this cost minimization, we introduce a novel criterion – a set of conditions, which if satisfied, guarantees global

optimality. In addition to this criterion, in Chapter 6, we also present a thorough convergence analysis for the novel algorithm we introduce. For this analysis, we follow the findings and arguments mentioned in [47]. It is important to note that, we discuss both stationary and non-stationary signal reconstruction in this section.

Finally, we complement this analysis by finding the best Regularization parameters out of a candidate set of values using a Model Selection approach. Traditional model selection criteria, such as Akaike Information Criteria (AIC) and Bayesian Information Criteria (BIC) are most effective for models estimated by maximum likelihood estimation, and therefore, cannot be directly applied for regularization parameter selection. Hence, we follow an information-theoretic approach for model selection, introduced by Shimamura et al. [76]. Their work is limited to the case of Group Lasso, where we extend it to the case of Sparse Group Lasso with "l_0 norm" approximation based regularization. Also, we introduce a novel basis for the stopping criterion of the algorithm. This criterion is based on the duality gap, where a dual function is formulated based on the primal cost function.

Since we are using an "l_0 norm" approximation model as the penalizer, we expect our algorithm to provide better reconstruction results compared to the currently used l_1 -norm methods, based on the required level of sparsity and the required level of measurements. Also, since we are embedding *a-priori* information in the structure of our algorithm, we expect the algorithm to perform well in a wide-range of problems

that fit this structure. In addition, most of the conventional reconstruction procedures use trial-and error to find the best set of Regularization parameters. We overcome this problem using an information-theoretic approach, and yield the best set of parameters among a set of candidate values. Finally, by the introduction of Duality and duality gap, we obtain a better measure for the stopping criterion in our algorithm.

2 COMPRESSIVE SAMPLING

In this chapter we initially introduce the idea behind Compressive Sampling, and then we define the Compressive Sampling problem. Next we discuss the importance of normed vector spaces and how it is related to defining the conditions for sparse recovery. Finally, we perform a theoretical analysis of structural sparsity and analyze its benefits.

2.1 Introduction to Compressive Sampling

Compressive Sampling or Compressive Sensing (CS) has gained much attention in the last decade in the fields of Signal Processing, Statistics, Computer Science, Applied Mathematics and Bio-Informatics. This is due to its ground-breaking ability to reconstruct signals from a far fewer set of samples/measurements as compared to conventional methods. Traditionally, the celebrated “Nyquist-Shannon Sampling Theorem”, introduced by Nyquist and Shannon, shows that a signal can be exactly recovered from a set of samples taken at the so-called Nyquist Rate (taken at twice the maximum frequency of the signal).

However, this stringent condition on sampling makes it impractical to cater to the ever-increasing need for reliable and fast sensing systems. For example, when dealing with signals with high bandwidths as in Radar or Ultra Wide Band signals, it is becoming difficult to acquire data at a rate of several GHz. Applications like seismic explorations, medical imaging applications such as Magnetic Resonance Imaging (MRI) and Functional Magnetic Resonance Imaging (fMRI) have constraints on the amount of sensors that can be used to acquire data. Also, to minimize the radiation exposure to patients, these real-time medical applications must have time-constraints in data-acquisition. These constraints would make data acquisition at the Nyquist rate very costly, time-consuming, or even infeasible.

Upon further analysis, it was evident that most of these signals of scientific interest are sparse/ compressible. In other words, it is possible that some or even most of the obtained data can be discarded without much perceptual loss; the useful information lies in a much smaller subspace compared to the overall signal space. This phenomenon brought the concept of *transform coding* to light and then later on the principle of *transform sparsity* in CS. *Transform Sparsity* states that, for a given sparse signal of interest $x = \left(x_i\right)_{i=1}^{n} \in \mathbb{R}^n$, there exists an orthonormal basis ψ such that $x = \psi\theta$ where θ being sparse. In general, this assumed level of sparsity can be described as: for some $C > 0 \text{ and } 0 < p < 2$.

$$\left\|\theta\right\|_p = (\sum_i \left|\theta_i\right|^p)^{\frac{1}{p}} \leq C \tag{2.1}$$

Therefore, when the signal is known to be sparse on its own or in a given basis, the minimum number of measurements required for "perfect" reconstruction of the sparse vector may become vastly reduced compared to that with the traditional Nyquist Sampling Theorem. The underlying idea behind CS is to directly capture the data in a compressed form rather than first sampling at a higher rate and then processing the sampled data and throwing away most of it. This allows data to be captured at a much lower sampling rate enabling a larger computational and sampling cost reduction.

2.2 The Compressive Sensing Problem

Let m, n represent the length of measurement vector and the length of the signal vector respectively. Let $A \in \mathbb{R}^{m \times n}$, $y \in \mathbb{R}^m$, $x \in \mathbb{R}^n$ be the gain matrix/sensing matrix, measurement/sensor vector and signal/source vector respectively. Due to the reduction in dimensionality, the number of measurements will be much less than the number of samples of the signal: $m << n$. Also, it is assumed that A does not include any zero columns. The CS problem can be stated as solving the following underdetermined linear system of equations to recover x, provided that x is sparse on its own or in a certain domain:

$$y = Ax \text{ or } y = A\psi\theta \tag{2.2}$$

The pioneering work of Candes, Romberg, Tao [29-32] and Donoho [33] revealed in their work that a sparsely represented signal can be recovered with a probability very close to one using a small set of linear, non-adaptive measurements.

This discovery enabled and encouraged the signal processing community to explore novel methods to recover the original signal from the compressive measurements. As a result, highly non-linear methods such as Convex Optimization, Combinatorial Algorithms and Greedy Algorithms emerged in CS as opposed to the computationally less demanding linear *sinc* interpolation signal recovery method used in the Nyquist-Shannon framework.

The reason for this is, unlike in the Nyquist case where the linear operator A can (in simple terms) be assumed as an $n \times n$ Identity matrix, in CS the operator becomes a highly "flat" matrix of $m \times n$, making unique recovery of the vector x or θ impossible. In order to understand the aforementioned signal recovery algorithms in CS, it is important to understand the key concepts in vector spaces and normed vector spaces.

2.3 Overview of Normed Vector Spaces and Justification for l_p norm

The l_p norm of a vector $x \in \mathbb{R}^n$ is defined as follows:

$$\left\|x\right\|_p = \begin{Bmatrix} \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}}, p \in (0, \infty) \\ \max_i |x_i|, \ p = \infty \\ |supp(x)|, \ p = 0 \end{Bmatrix} \tag{2.3}$$

The usual l_p norm where $p \geq 1$ is a convex function, and holds the triangle inequality - Fig. 2.1(a, b, c). However, when p is bounded such that $0 < p < 1$, l_p norm becomes highly non-convex as can be seen by the Astroid looking shape in Fig. 2.1(d). In other words, a line segment connecting any two points on the curve will lie above the curve, in a Euclidian space of at least two dimensions. Also, since the l_p norm $(0 < p < 1)$ fails to satisfy the triangle inequality, they are collectively referred to as *Quasi-norms*. "l_0 norm", which fails to satisfy many of the general norm properties like positive homogeneity (hence the quotation mark), merely denotes the cardinality of its support (number of non-zero). It can be denoted as follows:

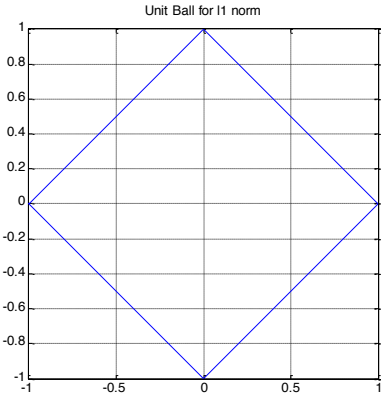
$$\|x\|_0 = |supp(x)| \tag{2.4}$$

$$\lim_{p \to 0} \|x\|_p^p = |supp(x)| \tag{2.5}$$

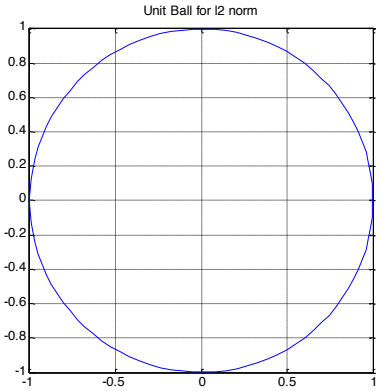
As mentioned earlier, the primary objective in CS is to obtain compressed data – m measurements, and to reconstruct the original sparse data using them. Most signals of scientific interest can be modeled as sparse signals, where they follow a power-law distribution. In other words, there are only a few significant coefficients, and the others can be equated to null without losing much information. To enforce sparsity onto the signal while reconstruction, l_p norm can be used. We can seek the sparsest solution of the underdetermine system $y = Ax$ as:

$$\hat{x} = \arg\min_x \|x\|_p \ \ s.t. \ y = Ax \tag{2.6}$$

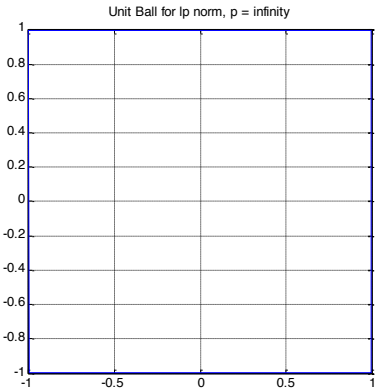
where $0 \leq p \leq 1$.



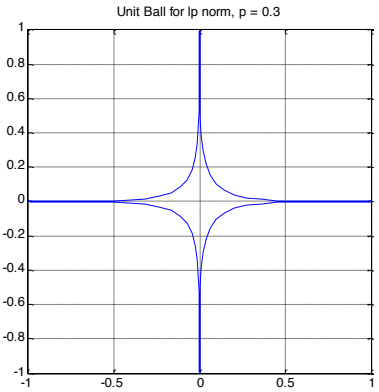
(a)



(b)



(c)



(d)

Figure 2.1: Unit balls for l_p norm, $p = 1, 2, \infty, 0.3, \|x\|_p = 1$

2.4 Conditions for Sparse Recovery

The conditions for sparse recovery are twofold. They are: conditions that need to be imposed on the sparsity of the original vector and the conditions on the sensing matrix

A. Let the minimization problems for l_0 and l_1 norm follow the equation (2.6) with $p = 0$ and 1 respectively. In this section we explain briefly the conditions for the uniqueness of l_0 norm regularization, the sufficient conditions for l_1 norm regularization solution to coincide with the l_0 norm solution, and the motivation, which propelled us to use l_0 norm based regularizers as opposed to its counterpart the l_1 norm. The necessary conditions for the l_0 norm solution to coincide with the l_1 norm solution are out of the scope of this explanation, but the interested reader is referred to [34, 35 - Chapter 7, 36, 37, 38] for detailed analysis and proof.

2.4.1 Uniqueness of l_0 norm based regularization

We first define the notion of "Spark", which stems from the terms "Sparse" and "Rank", and was introduced in [39].

Definition 2.1: A vector $x = \left(x_i\right)_{i=1}^{n}$ is called k – sparse if,

$$\left\|x\right\|_0 = \left|supp\left(x\right)\right| \le k$$

Definition 2.2: Let A be an $m \times n$ $(m < n)$ measurement matrix. $spark(A)$ is defined as the minimal number of linearly dependent columns of A.

In other words, if the rank of A is q, then $spark\left(A\right) = q + 1$. This is also referred to as A being rank $-q$ unambiguous [35].

Theorem 2.1: [39, 35 - Chapter 7] Let $A \in \mathbb{R}^{m \times n}, (m < n)$ be rank $-q$ unambiguous. For $p = 0$, if a solution of x for (2.6) is k – sparse, then $\hat{x}$ is a unique solution if and only if $k < \frac{spark(A)}{2} = (q+1)/2$.

From the definition of $spark(A)$, we can say, $spark(A) \in [2, m+1]$. Also we know that the maximum rank A could achieve is m. Therefore, Theorem 2.1 yields the requirement on the number of measurements as: $m \geq 2k$.

2.4.2 Sufficient Conditions for "$l_1 = l_0$"

a. ***Restricted Isometry Property (RIP):*** [31,41,42]

RIP property ensures that when the higher dimension source vector is projected to a lower dimension sensor vector using the flat A matrix, the information is still preserved.

Definition 2.3: Let A be an $m \times n$ $(m < n)$ measurement matrix. Then A has the RIP of order k, if there exists an $\alpha_k \in (0,1)$ s.t.

$$(1-\alpha_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1+\alpha_k)\|x\|_2^2 \text{ For all } x \in supp(x)$$

Theorem 2.2: [41, 42] Let A be a $m \times n$ $(m < n)$ measurement matrix and satisfies the RIP of order $2k$ with $\alpha_{2k} < \sqrt{2} - 1$. If x^* is the solution for (2.6) when $p = 1$, then

$$\left\| x - x^{*} \right\|_2 \leq C \frac{\sigma_k \left(x \right)_1}{\sqrt{k}}$$

In particular, if x is $k-$ sparse, the recovery is exact.

Where $\sigma_k \left(x \right)_1$ denotes the l_1 error of the best k – term approximation and C denotes a constant dependent on α_{2k}. The k – term approximation is the vector x with all but the k – largest entries set to zero.

Theorem 2.2 [42] states that the solution to the l_1 problem coincides with the solution to the l_0 problem provided that, $\alpha_{2k} < \sqrt{2} - 1.$ In other words, the convex relaxation is exact. Also, using this theorem, the error estimates for recovery from noisy data can be directly represented in terms of the best $k-$ term approximation. In [43], they show the best known RIP condition for sparse recovery using l_1 norm recovers all k – sparse vectors provided that A satisfies $\alpha_{2k} < 0.473.$

It is important to realize that even though spark and RIP provide guarantees to the recovery of $k-$ sparse signals, finding a matrix A which satisfies these properties is an NP hard problem. Due to this combinatorial computational complexity, more easily solvable property of Coherence of a matrix was introduced in [39].

b. ***Mutual Coherence:***

Definition 2.4: [40] For a given matrix $A = \left(a_i\right)_{i=1}^{n}$, where a_i is the i^{th} column of A, mutual coherence - $\mu(A)$ measures the smallest angle between each pair of its columns.

$$\mu\left(A\right) = \max_{i \neq j} \frac{\left|\left\langle a_i, a_j \right\rangle\right|}{\left\|a_i\right\|_2 \left\|a_j\right\|_2}$$

Theorem 2.3: Let A be a $m \times n \ (m < n)$ measurement matrix. For $p = 0$, if a solution of x for (2.6) satisfies,

$$\left\|x\right\|_0 < \frac{1}{2}\left(1 + \mu\left(A\right)^{-1}\right)$$

Then x is a unique solution of x for (2.6) for both $p = 0$ and $p = 1$.

In other words, if the sparsity of the solution to (2.6) satisfies the above mutual coherence condition of the measurement matrix, the solution for l_1 norm regularization will coincide with the solution for l_0 norm, and it will also be a unique solution.

Definition 2.5: Let A be an $m \times n (m < n)$ measurement matrix with rank-q unambiguity. Let $Null(A)$ denote the null space of A. Let $\delta \in Null\left(A\right), \left\|\delta\right\|_\infty = 1$ and $\tilde{\delta}$ be a sorted permutation of the absolute values of the coordinates δ_i of δ in a descending order s.t. $\tilde{\delta}_1 = \max_i \left|\delta_i\right| and \ \tilde{\delta}_n = \min\left|\delta_i\right|$. Let $S(A)$ be defined as $S\left(A\right) = \min \ \tilde{\delta}_{q+1}$ over all $\delta \in Null(A)$.

From the above definition it is important to notice that $0 < S(A) < 1$ [35-Chapter7].

Theorem 2.4: [35] Suppose A is rank $-q$ unambiguous, and x^* is the unique solution of x for (2.6) when $p = 0$. Let $\|x^*\|_0 = k$. If $k < \frac{(S(A)^p (q+1))}{(1+S(A)^p)}$ then the solution of x for (2.6) for any $p\,(0 \le p \le 1)$ is the same as x^*.

The above theorem reveals a significant discovery – for a given $p, 0 \le p \le 1$, the restriction for exact recovery becomes stricter as p increases. For $p = 0$, the above condition approaches $k < \frac{q+1}{2}$, which is exactly the necessary and sufficient condition for exact recover for l_0 norm regularization as stated in Theorem 2.1. Let us compare the level of sparsity required to satisfy the above condition for $p = 0.3$, $p = 0.5$ and $p = 1$, with $k_{0.3}$, $k_{0.5}$ and k_1 be the level of sparsity required to satisfy the above condition for the three p values, respectively. Then,

$k_{0.3} < \frac{S(A)^{0.3}(q+1)}{1+S(A)^{0.3}}$, $k_{0.5} < \frac{S(A)^{0.5}(q+1)}{1+S(A)^{0.5}}$, $k_1 < \frac{S(A)(q+1)}{1+S(A)}$. If we let $S(A) = 0.5$, then $k_{0.3} < 0.448(q+1)$, $k_{0.5} < 0.414(q+1)$ and $k_1 < 0.333(q+1)$. Hence, we can see that the upper bound for the level of sparsity decreases (stricter) as p increases from 0 to 1.

However, finding the global minimum for a l_p norm $(0 \le p \le 1)$ regularization problem is a difficult task. This is the key for our work, as we try to explore novel regularization

conditions to find the global optimum for l_p norm based regularizers where p is close to 0.

2.5 Theoretical analysis of Structural Sparsity and its benefits

As highlighted in Chapter 1, the structure of the problem plays a key-role in improving the quality of reconstruction. As mentioned previously, the popular Lasso method proposed by Tibshirani et al. (1996) [13] minimizes the usual sum of errors (least squares problem) with the l_1 norm regularization. In this section, we briefly introduce the concepts of Group Lasso and Sparse Group Lasso, which are popular extensions of Lasso that explore the structural sparsity of the problem. The Lasso cost function can be written as follows:

$$\min_x L(x) = \min_x \frac{1}{2}\|y - Ax\|_2^2 + \lambda \|x\|_1 \tag{2.7}$$

2.5.1 Group Lasso

Group Lasso was introduced by the statisticians Ming Yaun et al. [25] in order to improve the general factor selection of the input variables in the Lasso problem. Group Lasso tends to make selection based on the strength of groups of input variables as opposed to Lasso, which tends to make selection based on the strength of individual input variables.

Let x be divided into p non-overlapping groups such as:

$x = \left[x^1 x^2 \ldots . x^p \right]^T , x^l = \left[x_1^l x_2^l \ldots x_{n_l}^l \right]^T$ where x_i^l represents the i^{th} element of group l of the x vector. Let the length of a given group x^l be n_l, and therefore, $\sum_{i=1}^{p} n_i = n$.

Let A be divided into sub-matrices corresponding to the groups of x as follows:

$A = [A^1 A^2 \ldots A^p]$, where A^l is a m by n_l matrix. Let I_{n_l} represent the Identity matrix with dimensions $n_l \times n_l$. Therefore, the cost function for Group Lasso can be written as:

$$\min_x L(x) = \min_x \frac{1}{2} \left\| y - \sum_{l=1}^{m} A^l x^l \right\|_2^2 + \lambda \sum_{l=1}^{p} \sqrt{n_l} \left\| x^l \right\|_2 \tag{2.8}$$

It is important to note that the sparse penalty for the Group Lasso is a summation of $\left\| x^l \right\|_2$'s and not $\left\| x^l \right\|_2^2$'s. The latter penalty, also referred to as "Ridge Regression", is the l_2 norm of x and is everywhere differentiable. Therefore, it does not promote sparse solutions as discussed in Section 2.3. On the other hand, since $\sum_{l=1}^{p} \sqrt{n_l} \left\| x^l \right\|_2$ is not an l_2 norm of x, and is non-differentiable when $x = 0$, it may act as a shrinkage operator (ability to zero-out coefficients), making it a better sparse inducing penalty. From (2.8) it is evident that when $n_l = 1$ for all l, Group Lasso will reduce to just Lasso. It is also important to note that, Group Lasso uses the assumption that each A^l is orthonormalized, i.e. $\left(A^l \right)^T A^l = I_{n_l}$. This configuration of the penalty, having l_2 -norm

within the group and l_1 norm among the groups, encourages sparsity at group level, and hence has shown better results for group-like source reconstruction compared to Lasso.

A thorough theoretical analysis for Group Sparsity was done by J Huang et al. [27], where they introduced the definition of "Strong Group Sparsity - K_{GLasso}". They proved that for Group Lasso to be beneficial, the ratio $\frac{K_{GLasso}}{|supp(x)|}$ should be small. Also, they proved that under certain conditions, Group Lasso is more stable with respect to stochastic noise compared to Lasso. Most importantly, they showed that Group Lasso requires fewer number of measurement samples compared to Lasso for reconstruction.

Even though, Group Lasso sparsify among groups, it does not however yield sparsity within each group. In most applications, one would like to have both, sparsity among the groups as well as among the whole source vector. A similar example would be, as mentioned in the first chapter, the brain activations for different event related responses (ROI's), can be categorized as groups. This corresponds to the sparsity among the groups. But, during each event related response the brain might still not be fully "ON", and this corresponds to sparsity within each group. So for this kind of an experiment, the solution should be able to exploit both, sparsity within the group and among the groups.

2.5.2 Sparse Group Lasso

As a solution to the within group sparsity drawback in Group Lasso, Friedman et al. [58] introduced another l_1 norm penalty to the Group Lasso problem, which sparsify the whole source vector.

$$\min_x L(x) = \min_x \frac{1}{2}\left\| y - \sum_{l=1}^{p} A^l x^l \right\|_2^2 + \lambda_1 \sum_{l=1}^{p} \sqrt{n_l} \left\| x^l \right\|_2 + \lambda_2 \left\| x \right\|_1 \quad (2.9)$$

The above solution has shown to give superior reconstruction compared to Group Lasso, where both group-wise and within-group sparsity is present.

3 NON-CONVEX APPROACHES FOR "l_0 - norm" REGULARIZATION

3.1 Introduction to "l_0 - norm" based Regularization methods

As we mentioned earlier, "l_0 - norm" would be the best and the most natural choice for the sparsity inducing function - $F(x)$ because it imposes the least requirement on the sparsity k and therefore the least requirement on the size *m* of the samples to be collected. Using Lagrangian methods, we can re-write equation (2.6) as follows:

$$\hat{x} = \min_x \left[\left\| y - Ax \right\|_2^2 + \lambda F\left(x\right) \right] \tag{3.1}$$

Where λ is the Regularization parameter

$$F\left(x\right) = \left\| x \right\|_0 \tag{3.2}$$

Given that x is a k- sparse vector ($|supp(x)| \leq k$), in order to find an optimum x, one has to do combinatorial search over all possible k- sparse vectors satisfying $y = Ax$. This is an NP-Hard problem, where the computation cost will increase exponentially as the dimension of x increases.

Alternatively, as we discussed before, we can use l_p norm $(0 < p < 1)$ methods or continuous functions which mimic the behavior of the l_0 norm. Even though these continuous functions are highly non-convex in nature, convex relaxation forms of them [44-46] need a considerably less amount of measurements and computation to exactly recover the sparse signal than the l_1- norm. Some popular l_1- norm minimization methods include the Basis Pursuit (BP) [48-50], while the l_p norm $(0 < p < 1)$ methods include FOCUSS [12].

3.1.1 Non-convex sparsity inducing functions

The sparsity inducing objective functions can be generally expressed as follows:

$$F_\sigma(x) = n - \sum_{i=1}^{n} f_\sigma(x_i), \quad x \in \mathbb{R}^n \tag{3.3}$$

Most non-convex functions that approximate the "l_0- norm" are collectively non-decreasing functions, which enjoy the following properties [46, 47, 51]:

a. $\lim_{\sigma \to 0} f_\sigma(x_i) = 0 \; ; x_i \neq 0$

b. $f_\sigma(0) = 1$

c. $f_\sigma(x_i)$ has a continuous and bounded derivative for $x_i \in (0, \infty)$

d. Its derivative $f_\sigma'(x_i)$ satisfies,

$$\left\{ \begin{array}{ll} f_\sigma'(x_i) \to 0 & \text{for } x_i \gg \sigma \\ f_\sigma'(x_i) \text{ is a large value for } x_i \ll \sigma & \end{array} \right\}$$

Function Name	$f_\sigma(x_i)$
SL0	$e^{\frac{-x_i^2}{2\sigma^2}}$
atan	$\operatorname{atan}\left(\frac{\lvert x_i\rvert}{\sigma}\right)$
log-sum	$\frac{1}{\log(1+\sigma)}\log\left(\lvert x_i\rvert+\sigma\right)$
log-exp	$\frac{1}{\log 2}\log\left(\frac{2}{1+e^{-\lvert x_i\rvert/\sigma}}\right)$
triangular	1 if $\lvert x_i\rvert \geq \sigma$ $\frac{\sigma+x_i}{\sigma}$ if $-\sigma \leq x_i \leq 0$ $\frac{\sigma-x_i}{\sigma}$ if $0 \leq x_i \leq \sigma$
truncated hyperbolic	1 if $\lvert x_i\rvert \geq \sigma$ $1-\left(\frac{x_i}{\sigma}\right)^2$ if $\lvert x_i\rvert < \sigma$

Table 3.1: Some of the well-known Non-convex Sparsity inducing functions [46, 47, 51]

In other words, $(1-f_\sigma(x_i))$ is a uni-variate function which approximates the Kronecker Delta function - $\delta_{x_i,0}$ and σ determines the quality of the approximation. Therefore $n-\sum_{i=1}^{n} f_\sigma(x_i)$ approximates $\|x\|_0$, and approaches $\|x\|_0$ when σ approaches 0.

The following figures demonstrate the behavior of some of the functions in Table 3.1.

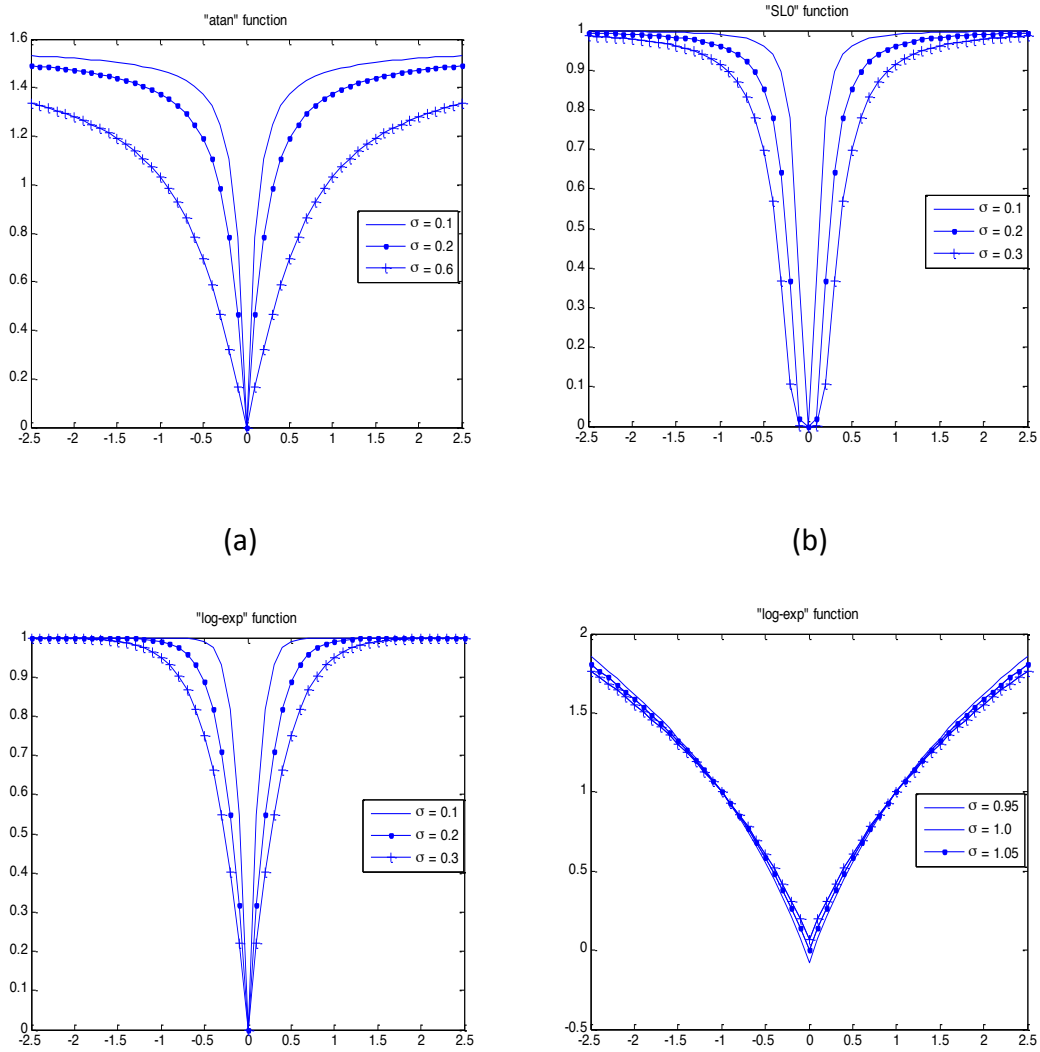


Figure 3.1: Non-convex Sparsity inducing functions for different σ values

3.1.2 Introduction to the 'SL0' method

In this work we use the Gaussian based function – SL0, due to its simplicity, and the abundance of previous literature [46, 47, 54, 55] pertaining to it. The advent of the SL0 method provided a faster algorithm compared to the basis pursuit (BP) method and an

improved quality of estimation compared to the BP, FOCUSS and Matching Pursuit (MP) [52, 53] methods.

The SL0 Minimization model:

$$L_{SL0}(x) = \frac{1}{2}\|y - Ax\|_2^2 + \lambda\ (n - \sum_{i=1}^{n} e^{\frac{-x_i^2}{2\sigma^2}}) \tag{3.4}$$

$$\hat{x} = \arg\min_{x} \left[\frac{1}{2}\|y - Ax\|_2^2 + \lambda\ (n - \sum_{i=1}^{n} e^{\frac{-x_i^2}{2\sigma^2}}) \right] \tag{3.5}$$

As seen from Figure 3.1(b), σ plays an important role in the approximation function. It determines the convexity of the function. It is evident that for small values of $\sigma, F_\sigma(x)$ tend to be highly peaky, well approximates $\|x\|_0$. However, in such a case it is also highly non-smooth, and the function could potentially contain many local minima. The convexity of this function depends on both the signal values and the value for σ. We will explain this in more detail in Section 3.2.

Having many minima adversely affects the global optimality of the minimization model using a gradient based algorithm. To circumvent this problem, the authors of SL0 introduced a clever method of using the idea of Graduated Non-Convexity (GNC) – a deterministic form of Simulated Annealing [56]. The non-randomness will save much computational time during the minimization process. The idea is to start the minimization (3.5) with a large σ, and find the optimal point using a gradient based method. Next, using the previous optimal point as the initial condition, with a reduced σ

, a new optimal point is found again. In general, an outer loop will decrement the σ value, while the inner loop will use the gradient descent to find the optimal solution for the given σ value.

When $\sigma \to \infty$, from (3.5), $\hat{x}$ admits a closed-form solution because the sparsity inducing term $F_\sigma(x)$ becomes zero. In fact, this solution is the least squares solution, which can be written as follows:

$$\lim_{\sigma \to \infty} \hat{x} = \left(A^T A \right)^{-1} A^T y \tag{3.6}$$

Since the cost function L_{SL0} is convex when $\sigma \to \infty$, there will only be one minimum, which can be demonstrated by the lowermost curve in the Figure 3.2. This solution will be used as the initial approximation for all the SL0 based algorithms [46, 47]. Subsequently, σ is reduced by a small amount and the solution for equation (3.5) is solved again, taking the previous global minimum as the initial condition (as shown in Figure 3.2). Following the concept of GNC [46, 47, 56 – Chapter 3,7], this procedure is repeated until a stopping criterion is satisfied (This stopping criterion will be explained later in Chapter 6). As illustrated in Figure 3.2, it is expected that the minimization of the corresponding cost functions for decreasing σ will eventually end up at the global minimum.

The concept of GNC is vastly used in the literature pertaining to non-convex based "l_0 norm" approximation models to avoid the solution to be trapped in local minima [46, 51]. Even though Blake et al. [56] proved the global convergence properties for GNC for an Energy function based on weak-string and membrane, the global convergence for a general and an arbitrary non-convex function has not yet been proven. In our work (described in Chapter 4), we introduce a novel criteria for the case of "SL0 based Sparse Group penalization", which if satisfied, will guarantee the cost function to be convex, therefore yielding a global minimum. This criterion is checked for the smallest σ, therefore, if satisfied, the global optimum can be found without having to iterate over a sequence of σ's.

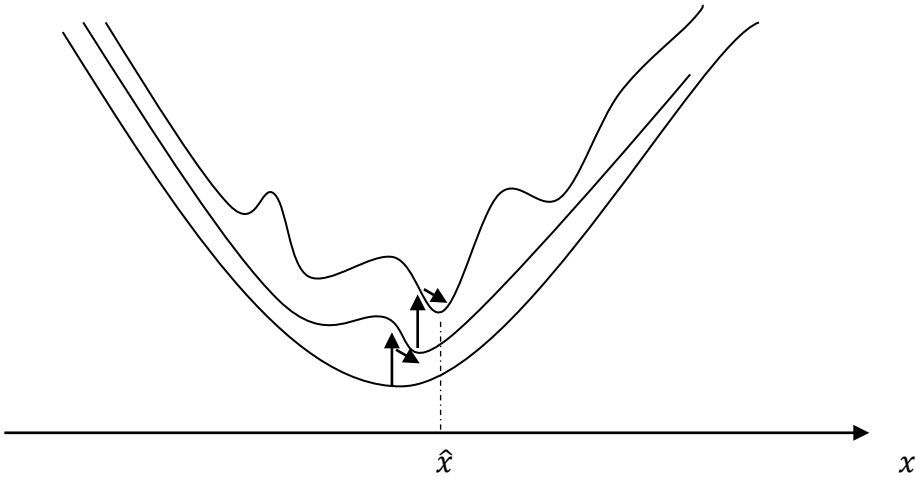


Figure 3.2 Concept of Graduated Non-Convexity

From the work carried out in [46], Mohimanni et al. has shown that the SL0 method recovers the sparse signal exactly with faster convergence and comparably less amount of measurements. If $f_\sigma(x_i)$ is chosen such that it follows the properties stated in

Section 3.1.1, then Theorem 1 in [46] states that, for a Measurement matrix A, which satisfies the Unique Representation Property (URP) [78] ,

$$\lim_{\sigma \to 0} \hat{x} = x^* \tag{3.7}$$

Here $\hat{x}$ represents the reconstructed SL0 solution, while x^* represents the actual unique sparse solution. (The URP property states that for a given matrix $A \in \mathbb{R}^{m \times n}$, every $m \times m$ sub-matrix is invertible, i.e. A is rank-m unambiguous)

Later works of Mohimani et al. in [47] provides a detailed convergence analysis for the GNC?. Even though this work gives a thorough analysis of the behavior of σ, and provides a global optimality criterion, to satisfy this criterion, it requires finding the Asymmetric Restricted Isometry Constants (ARIC) of the dictionary A. Precise calculation of ARIC involves enumerating through all possible n_0 column sub-matrices of the dictionary A, and then computing their smallest singular values (Definition for n_0 can be found in [47]). Hence, when the dimensions of the problem increase, the complexity grows exponentially making it intractable to find the ARIC's.

3.2 Theoretical analysis of the non-convexity of 'SL0' method

In order to guarantee a globally optimal solution, a sufficient condition is that the cost function in the minimization model should be convex. One popular method of proving convexity of a function is to show that its Hessian is positive definite. The conditions for

a positive definite matrix H can be found using the Gershgorin's circle theorem [57], the Sylvester's Criterion or checking the positivity of $z^T Hz$ (where z is any non-zero real vector and z^T being its transpose).

Upon careful analysis, we found that in-order to force positive definiteness using the first two methods; we would have to assume diagonal dominance in the Hessian of the cost function – H_{SL0} Since this is not a valid assumption, we used the criteria on positivity for $z^T H_{SL0} z$ in our approach.

Let's assume $x \in \mathbb{R}^n, y \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$ to be the sparse signal vector, measurement vector, and the measurement matrix, respectively. From equation (3.4), the gradient of L_{SL0} can be written as follows,

$$\frac{\partial L_{SL0}}{\partial x} = -A^T \left(y - Ax \right) + \frac{\lambda W\left(x\right) x}{\sigma^2} \tag{3.8}$$

where $W\left(x\right) = \begin{bmatrix} e^{\frac{-x_1^2}{2\sigma^2}} & 0 & 0 \\ 0 & . & 0 \\ 0 & 0 & e^{\frac{-x_n^2}{2\sigma^2}} \end{bmatrix}$ is an $n \times n$ diagonal matrix.

Therefore, the Hessian can be written as:

$$\frac{\partial L_{SL0}^2}{\partial x \partial x^T} = A^T A + \frac{\lambda}{\sigma^2}\left[W(x) - \frac{W(x)\tilde{X}(x)}{\sigma^2}\right] = H_{SL0} \stackrel{?}{>} 0 \tag{3.9}$$

where $\tilde{X}(x) = \begin{bmatrix} x_1^2 & 0 & 0 \\ 0 & . & 0 \\ 0 & 0 & x_n^2 \end{bmatrix}$ is an $n \times n$ diagonal matrix.

If H_{SL0} is positive definite, the cost function L_{SL0} would have only one local minimum to converge to. Since $n > m$, A will not be full column rank. Therefore, $A^T A$ is a singular matrix. In other words, if we check the positive definiteness of $A^T A$:

$$z^T (A^T A) z \stackrel{?}{>} 0 \tag{3.10}$$

$$z^T (A^T A) z = \|Az\|_2^2 \geq 0$$

It is obvious that the norm of a vector is non-negative, and hence the fidelity term $\frac{1}{2}\|y - Ax\|_2^2$ is positive-semi definite. This is unfortunate, since it doesn't render us the opportunity to force the Hessian to be positive at all times.

Therefore, now we look at the positive definiteness of the sparsity inducing term:

Since $\lambda, \sigma > 0$, we can check the positivity of the following,

$$z^T\left[W\left(x\right)-\frac{W\left(x\right)\tilde{X}\left(x\right)}{\sigma^2}\right]z>^?0 \tag{3.11}$$

$W(x)$ is a diagonal matrix with exponential entries – i.e. it's a diagonal matrix with positive entries. Therefore, we can re-write the above requirement as,

$$z^T\left[I_n-\frac{\tilde{X}\left(x\right)}{\sigma^2}\right]z>^?0$$

Where I_n is an $n\times n$ Identity matrix.

Since $\tilde{X}\left(x\right)$ is also a diagonal matrix with positive entries, if all the Eigen values in $\left[I_n-\frac{\tilde{X}(x)}{\sigma^2}\right]$ are positive, then the Hessian of the sparsity inducing term will be positive definite. Therefore, for the cost function L_{SL0} to be strictly convex, for all $i(i\in[1,n]),\ 1>x_i^{\,2}/\sigma^2$. This gives us the relationship between the signal and the value σ in order to guarantee a global minimum for L_{SL0}. As long as $\sigma^2>x_i^2$, for all $(i\in[1,n])$, the cost function will be guaranteed to converge to a global minimum. But, it is obvious that when σ is decreased gradually, this condition would eventually be violated. This asserts us that the cost function may eventually be subjected to many local minima as σ is decreased.

4 METHODOLOGY

With the background presented in the previous two chapters, this chapter is devoted to finding algorithms to solve the source location problem using the GNC method to minimize some SL0 based cost functions. We pay particular attention to the important and yet difficult global convergence issue.

In Section 4.1, as a part of our contribution, we will initially introduce a novel method to solve the non-convex optimization problem of SL0 using a quadratic majorizer approach. Next, in Section 4.2, we will extend the problem to the Sparse Group version of SL0. We name it as "Sparse Group SL0 – SGSL0", which to our knowledge is an original contribution. Also, in this section, we introduce a novel Global Optimality criterion using the positive definite properties of the Hessian function of the cost model in SGSL0. If this criterion is satisfied, the cost function in SGSL0 will be convex, and therefore, would guarantee a global minimum. Section 4.3 discusses the method of optimization and the development of the novel algorithm SGSL0.

Finally, Section 4.4 is devoted to the time varying signal reconstruction case. This will extend the sensor and source vectors to the corresponding sensor and source matrices where the columns will represent each time sample.

4.1 Method of Optimization for SL0

The method used in this section is convex relaxation using a quadratic majorizer. The idea of using a quadratic model as the majorizer is to replace the gradient descent by a 2nd order Taylor approximation. This can also be interpreted as iteratively minimizing the cost function locally using the 2nd order Taylor approximation. This procedure can be considered as a special case of majorizer-minimzation technique, which will be explained in more detail in Section 4.3.1. A similar approach was used by Monetefusco et al. [51], where they used a local linear approximation function (1st order Taylor approximation) for the same purpose. This type of a minimization procedure is synonymous with the proximal gradient method. Since the 2nd order Taylor approximation is used instead of the first order approximation as in the normal gradient descent method, the approximation is improved during the descending process, and therefore it will have a faster convergence rate.

Let's re-write the SL0 cost model as follows:

$$L_{SL0}(x) = \frac{1}{2}\|y - Ax\|_2^2 + \lambda\ (n - \sum_{i=1}^{n} e^{-x_i^2/2\sigma^2}) \tag{4.1}$$

Let $h(x) = \frac{1}{2}\|y - Ax\|_2^2$ and $g(x) = (n - \sum_{i=1}^{n} e^{-x_i^2/2\sigma^2})$. Therefore, minimization model of Equation (4.1) can be written as,

$$\min_{x} L_{SL0} = \min_{x} \left[h(x) + \lambda g(x) \right]$$

As we previously discussed in Chapter 3, in order to obtain the global convergence we will be using the Graduated Non-Convexity method (refer to Section 3.1.2), where we would use a decrementing sequence of σ values throughout the minimization process. Let's assume that for a large enough σ, $g(x)$ is convex. Assuming that the current iterate for $x : x^{(k-1)}$ lies in close proximity to the next iterate $x^{(k)}$, we use the Quadratic approximation - $q(x)$ to iteratively minimize $g(x)$ as follows:

$$q\left(x^{(k)}\right) = g\left(x^{(k-1)}\right) + \nabla g\left(x^{(k-1)}\right).\left(x^{(k)} - x^{(k-1)}\right) + \frac{L}{2}\left\|x^{(k)} - x^{(k-1)}\right\|_2^2 \tag{4.2}$$

where $\nabla g(x)$ represents the gradient of the $g(x)$ function. It is important to note that, the constant $L(L \geq 0)$ can be chosen such that, $q(x^{(k)})$ will act as a surrogate function to $g(x^{(k)})$ (See Section 4.3.1 for a detailed discussion on surrogate functions). This constant L will be chosen using a backtracking line search at every iteration of k, i.e., the smallest L will be chosen which will satisfy the following criterion:

$$q\left(x^{(k)}\right) \geq g\left(x^{(k)}\right) \tag{4.3}$$

Now let's consider the minimization of $q(x^{(k)})$ with respect to $x^{(k)}$. We can include the gradient term inside the quadratic term and re-write equation (4.2) as:

$$\min_{\mathrm{x}^{(\mathrm{k})}} q\left(\mathrm{x}^{(\mathrm{k})}\right) = \min_{\mathrm{x}^{(\mathrm{k})}} \left[\frac{L}{2} \left\| x^{(k)} - x^{(k-1)} + \frac{\nabla g\left(x^{(k-1)}\right)}{L} \right\|_2^2 - \left\| \frac{\nabla g\left(x^{(k-1)}\right)}{2\mathrm{L}} \right\|_2^2 + g\left(x^{(k-1)}\right) \right]$$

After removing the constant terms we get,

$$\min_{\mathrm{x}^{(\mathrm{k})}} q\left(x^{(k)}\right) = \min_{\mathrm{x}^{(\mathrm{k})}} \frac{L}{2} \left\| x^{(k)} - \left(x^{(k-1)} - \frac{1}{L} \nabla g\left(x^{(k-1)}\right) \right) \right\|_2^2 \tag{4.4}$$

From equation (3.8) we can get, $\nabla g\left(x^{(k-1)}\right) = \dfrac{W\left(x^{(k-1)}\right) x^{(k-1)}}{\sigma^2}$

Therefore from equation (4.1), the total cost function can be minimized by

$$\min_{x^{(k)}} L_{SL0}\left(x^{(k)}\right) \approx \min_{x^{(k)}} \left[\frac{1}{2} \left\| y - Ax^{(k)} \right\|_2^2 + \left(\frac{\lambda L}{2} \right) \left\| x^{(k)} - \left(x^{(k-1)} - \frac{1}{L} \nabla g\left(x^{(k-1)}\right) \right) \right\|_2^2 \right]$$

$$\frac{\partial L_{SL0}}{\partial x^{(k)}} = -A^T\left(y - Ax^{(k)}\right) + \lambda L\left[x^{(k)} - x^{(k-1)} + \frac{1}{L} \nabla g\left(x^{(k-1)}\right) \right] = \vec{0}$$

Therefore, the update equation can be written as,

$$x^{(k)} = \left(A^T A + \lambda L I_n \right)^{-1} \left[\lambda L x^{(k-1)} - \lambda \nabla g\left(x^{(k-1)}\right) + A^T y \right] \tag{4.5}$$

Here I_n represents the $n \times n$ identity matrix.

It is important to note that the matrix inversion term $\left(A^T A + \lambda L I_n \right)^{-1}$ is only required to compute once through-out the iterative process, hence saving computing time.

Algorithm 1 – Quadratic majorizer based SL0 (QSL0)

1. Input : $\lambda, \left[\sigma_{max} \ldots \sigma_{min}\right], A \in \mathbb{R}^{m \times n}, \; y \in \mathbb{R}^{m}$

2. Initialization : Using Minimum Norm Solution :

$x_0 = (A^T A)^{-1} A^T y$

3. *for* $\sigma_{\max} : \sigma_{\min}$ *do*

4. *for* $k = 1 : k_{\max}$

5. Find L which satisfies (4.3) using a backtracking line search algorithm

6. Update x using the following:

7. $x^{(k)} = (A^T A + \lambda L I_n)^{-1}\left[\lambda L x^{(k-1)} - \lambda \nabla g(x^{(k-1)}) + A^T y\right]$

8. *end for*

9. *end for*

(The simulation results for this algorithm is included in Section 5.1 in Chapter 5)

4.2 Global Optimality conditions for "Sparse Group SL0 – SGSL0"

We now turn out attention to the concept of group LASSO, in particular the sparse group LASSO as we discussed earlier. We first investigate a novel global optimality condition for SGSL0. We follow the same notations described in Section 2.5.1. The

Sparse Group Lasso replaced by the "l_0 - norm" approximation will be developed as follows:

$$\min_{x} L_{SGSL0}(x) = \min_{x}\left[\frac{1}{2}\left\|y-\sum_{l=1}^{p}A^{l}x^{l}\right\|_2^2+\lambda_1\sum_{l=1}^{p}\sqrt{n_l}\left\|x^{l}\right\|_2+\lambda_2\left\|x\right\|_0\right] \tag{4.6}$$

But, $\left\|x\right\|_0=\sum_{l=1}^{p}\left\|x^{l}\right\|_0$

Therefore,

$$\min_{x} L_{SGSL0}(x) = \min_{x}\left[\frac{1}{2}\left\|y-\sum_{l=1}^{p}A^{l}x^{l}\right\|_2^2+\lambda_1\sum_{l=1}^{p}\sqrt{n_l}\left\|x^{l}\right\|_2+\lambda_2\sum_{l=1}^{p}\left\|x^{l}\right\|_0\right] \tag{4.7}$$

Substituting the "l_0 norm" approximation $g(x)$ and also for simplicity letting each group to have an equal length of n_l (i.e. to merge $\sqrt{n_l}$ into λ_1 in the 2nd term), the above can be re-written as:

$$\min_{x} L_{SGSL0(x)} = \left[\frac{\lambda_0}{2}\left\|y-\sum_{l=1}^{p}A^{l}x^{l}\right\|_2^2+\lambda_1\sum_{l=1}^{p}\left\|x^{l}\right\|_2+\lambda_2\sum_{l=1}^{p}\left[n_l-\sum_{i=1}^{n_l}e^{\frac{-(x_i^l)^2}{2\sigma^2}}\right]\right] \tag{4.8}$$

Here, $\lambda_0,\lambda_1,\lambda_2 \geq 0$ are the corresponding regularization/ tuning parameters for each term. Let's consider the behavior of the three terms of the above minimization problem, respectively, minimized with respect to x (and x^l).

$$\frac{1}{2}\left\|\boldsymbol{y}-\sum_{l=1}^{p}\boldsymbol{A}^{l}\boldsymbol{x}^{l}\right\|_2^2:$$

This is called the data fidelity term, and will be used to define the solution domain irrespective of sparsity. This function is a convex function, but since the Gain matrix A has more columns than rows: $m << n$; at most it can only be a full row rank matrix. Therefore, it will only be convex but not strictly convex. Hence, its Hessian will be positive semi-definite and not positive definite when minimized with respect to x.

In other words, the Hessian of the fidelity term will be $(A)^T A$, which is an $n \times n$ matrix, and therefore, it is singular.

Now let us consider minimizing the above fidelity term with respect to x^l. We will assume that the A matrix is full row-rank and that the maximum length for a group does not exceed the number of rows: $m \geq \max(n_l)$.

Correspondingly, the Hessian for the fidelity term with respect to x^l would be,

$$\frac{\nabla^2 \left[\frac{1}{2} \left\| y - \sum_{l=1}^{p} A^l x^l \right\|_2^2 \right]}{\nabla \left(x^l \right) \nabla \left(x^l \right)^T} = \left(A^l \right)^T A^l \tag{4.9}$$

Since $\left(A^l \right)^T A^l$ is a matrix of dimension $n_l \times n_l$, it has full rank and is non-singular.

Therefore, we can conclude that $\left(A^l \right)^T A^l$ is a positive definite matrix. In other words, all of $\left(A^l \right)^T A^l$'s Eigenvalues will be positive.

$\sum_{l=1}^{p} \left\| \boldsymbol{x}^l \right\|_2$:

The minimization of this with respect to x^l would be the sub-gradient of the l_2 norm of x^l:

$$\frac{\partial \sum_{l=1}^{p} \left\| x^l \right\|_2}{\partial x^l} = \frac{\partial \left\| x^l \right\|_2}{\partial x^l} = \begin{cases} \dfrac{x^l}{\left\| x^l \right\|_2} & for\ x^l \neq \vec{0} \\ \left\{ Z : \left\| Z \right\|_2 \leq 1 \right\} & for\ x^l = \vec{0} \end{cases}$$

It is important to note here that Z can be any real vector where $\left\| Z \right\|_2 \leq 1$. The reason for this kind of a behavior for the gradient of $\left\| x^l \right\|_2$ is its discontinuity at $x^l = \vec{0}$.

Now let us consider the Hessian of $\left\| x^l \right\|_2$ for $x^l \neq \vec{0}$:

$$\frac{\partial^2 \left\| x^l \right\|_2}{\partial x^l \left(\partial x^l \right)^T} = \frac{1}{\left\| x^l \right\|_2^3} \left[\left\| x^l \right\|_2^2 I_{n_l} - x^l \left(x^l \right)^T \right] \tag{4.10}$$

Here, I_{n_l} represents the Identity matrix with dimensions $n_l \times n_l$.

The Hessian, for $x^l = \vec{0}$ will be a matrix of $n_l \times n_l$ with all the elements being 0.

$$\sum_{l=1}^{p} \left[n_l - \sum_{i=1}^{n_l} e^{-\frac{-(x_i^l)^2}{2\sigma^2}} \right] \ (\text{As } \sigma \to 0):$$

This quasi-convex function will have the following gradient with respect to x^l:

$$\frac{\nabla\left[\sum_{l=1}^{p}\left(n_l-\sum_{i=1}^{n_l}e^{\frac{-\left(x_i^l\right)^2}{2\sigma^2}}\right)\right]}{\nabla\left(x^l\right)}=-\frac{\nabla\left(\sum_{i=1}^{n_l}e^{\frac{-\left(x_i^l\right)^2}{2\sigma^2}}\right)}{\nabla\left(x^l\right)}=\frac{W\left(x^l\right)x^l}{\sigma^2}$$

Likewise, the Hessian would be:

$$-\frac{\nabla^2\left(\sum_{i=1}^{n_l}e^{\frac{-\left(x_i^l\right)^2}{2\sigma^2}}\right)}{\nabla\left(x^l\right)\nabla\left(x^l\right)^T}=\frac{1}{\sigma^2}\left[W\left(x^l\right)-\frac{W\left(x^l\right)\tilde{X}\left(x^l\right)}{\sigma^2}\right] \tag{4.11}$$

Now using (4.9), (4.10) and (4.11), the Hessian for the total cost function with respect to x^l can be written as follows:

$$\frac{\partial^2 L_{SGSL0}\left(x\right)}{\partial\left(x^l\right)^T\partial\left(x^l\right)}=\lambda_0\left(A^l\right)^T A^l+\frac{\lambda_1}{\left\|x^l\right\|_2^3}\left[\left\|x^l\right\|_2^2 I_{n_l}-x^l\left(x\right)^T\right]+\frac{\lambda_2}{\sigma^2}\left[W\left(x^l\right)-\frac{W\left(x^l\right)\tilde{X}\left(x^l\right)}{\sigma^2}\right] \tag{4.12}$$

In (4.12), the first term is the Gram Matrix of A^l, and is positive definite as discussed earlier. In other words, all its Eigenvalues are positive. For the worst case scenario let us consider the minimum Eigenvalue of $\left(A^l\right)^T A^l$ to be q_l i.e. $\left(A^l\right)^T A^l \geq q_l I_{n_l}$.

The 2nd term is the Hessian for l_2 norm, and its positive definiteness can be checked as follows:

For a given column vector $v \in \mathbb{R}^{n_l}$:

$$v^T\left[\left\|x^l\right\|_2^2 I_{n_l}-x^l\left(x^l\right)^T\right]v \;\overset{?}{>}\; 0$$

$$v^T \left\| x^l \right\|_2^2 I_{n_l} v - v^T x^l \left(x^l \right)^T v \quad >^? 0$$

$$v^T \left\| x^l \right\|_2^2 I_{n_l} v - < \left(x^l \right)^T v >^2 \quad >^? \quad 0$$

$$\left\| v \right\|^2 \left\| x^l \right\|^2 \quad >^? \quad < \left(x^l \right)^T v >^2$$

But, the above inequality is the same as the Cauchy-Schwarz inequality which is always true. Therefore, we can conclude that

$$v^T \left[\left\| x^l \right\|_2^2 I_{n_l} - x^l \left(x^l \right)^T \right] v \geq 0$$

In other words, the 2nd term of equation (4.12) is a positive semi-definite matrix. In the worst case scenario, the Eigenvalues of this Hessian can be zero. Hence, this term cannot be used to guarantee convexity on the overall cost function L_{SGSL0}.

Now let us consider the 3rd term of (4.12). Since $W(x^l)$ is a positive definite matrix, the 3rd term is an addition of a convex and a concave term. The convexity or concavity is determined by the parameter σ^2. The worst case scenario is when σ^2 is small enough that it forces the following entire term to be a negative value.

$$v^T \left[W\left(x^l \right) - \frac{W\left(x^l \right) \tilde{X}\left(x^l \right)}{\sigma^2} \right] v \tag{4.13}$$

This can be expanded as,

$$v^T \left(\begin{bmatrix} e^{\frac{-x_1^l}{2\sigma^2}} & 0 & 0 \\ 0 & . & 0 \\ 0 & 0 & e^{\frac{-x_{n_l}^l}{2\sigma^2}} \end{bmatrix} - \frac{1}{\sigma^2} \begin{bmatrix} (x_1^l)^2 e^{\frac{-x_1^l}{2\sigma^2}} & 0 & 0 \\ 0 & . & 0 \\ 0 & 0 & (x_{n_l}^l)^2 e^{\frac{-x_{n_l}^l}{2\sigma^2}} \end{bmatrix} \right) v$$

Therefore the above term can be re-written as,

$$v^T W\left(x^l\right)\left(I_{n_l} - \frac{1}{\sigma^2}\begin{bmatrix}(x_1^l)^2 & 0 & 0\\ 0 & . & 0\\ 0 & 0 & (x_{n_l}^l)^2\end{bmatrix}\right)v$$

Let the maximum value of x^l be $x^l{}_{\max}$ therefore,

$$(x_{max}^l)^2 I_{n_l} \geq \begin{bmatrix}(x_1^l)^2 & 0 & 0\\ 0 & . & 0\\ 0 & 0 & (x_{n_l}^l)^2\end{bmatrix}$$

Hence, for the worst case scenario, Hessian for the convex-concave part in equation (4.12) can be written as,

$$\frac{\lambda_2 W\left(x^l\right)}{\sigma^2}\left(I_{n_l} - \frac{\tilde{X}\left(x^l\right)}{\sigma^2}\right) \geq \frac{\lambda_2 W\left(x^l\right)}{\sigma^2}\left(1 - \frac{(x_{max}^l)^2}{\sigma^2}\right)I_{n_l}$$

The largest Eigen value of $W\left(x^l\right)$is 1, when $x_i^{\,l} = 0$.

Therefore, considering all of the above, for the worst case scenario (when concavity is at its maximum effect on the overall Hessian), the positive definiteness of the overall Hessian can be checked by,

$$v^T\left[\lambda_0 q_l I_{n_l} + \frac{\lambda_2}{\sigma^2}\left(1 - \frac{(x_{max}^l)^2}{\sigma^2}\right)I_{n_l}\right]v \geq^{?} 0$$

Therefore, if

$$\lambda_0 q_l + \frac{\lambda_2}{\sigma^2}\left(1 - \frac{(x_{max}^l)^2}{\sigma^2}\right) \geq 0 \qquad (4.14)$$

is satisfied, the overall cost function with respect to x^l will be convex, and therefore the minimization will guarantee the global minimum. This is our global optimality condition, and it is dependent on the regularization parameters, the σ value, and also a maximum possible value for x^l. In the next section we will present a strategy to guarantee global convergence iteratively by verifying against this Global Optimality condition. (Verify if this statement is true or not)

4.3 Method of Optimization for SGSL0

In this section we present a method of optimization and its associated algorithm for the sparse group LASSO. As discussed earlier, due to the non-differentiability of the SGSL0 cost function, we cannot use a simple gradient search method. Instead, the method presented here is based on the proximal gradient method.

4.3.1 Proximal Gradient method

Proximal Gradient method [65, 67] is specifically tailored to optimize a cost function, which has a combination of a differentiable-smooth function with L Lipschitz constant and a non-smooth function. Lipschitz condition is an important property for smooth functions, which assures continuity of the function. For a given smooth function $u(x)$, the Lipschitz constant is defined as the smallest $L \geq 0$ for all x_1, x_2 in x which satisfies,

$$\left\|\nabla u\left(x_1\right)-\nabla u\left(x_2\right)\right\| \leq L\left\|x_1-x_2\right\|$$

This is also referred to as $\nabla u\left(x\right)$ being Lipschitz continuous with constant L.

Proximal Gradient method has gained much attention in the recent years, due to its faster convergence rates, and the ability to work with large number of data. Since it is a first-order method, which uses only the gradient information, it is significantly more scalable than the conventional Interior point methods for the conventional Second Order Cone Programming (SOCP) techniques. An accelerated version of this method (discussed in the Fast Iterative Shrinkage-Thresholding Algorithm –FISTA [66]) has a convergence rate of $O\left(\frac{1}{\varepsilon}\right)$ for a desired accuracy ε, which is much faster than a rate of $O\left(\frac{1}{\varepsilon^2}\right)$ for the standard sub-gradient method. Computing the exact proximal operator efficiently is the key to enjoying these perks using this method.

Let us consider the minimization problem given below:

$$\min_x h\left(x\right)=\min_x\left[u\left(x\right)+v\left(x\right)\right] \tag{4.15}$$

where, $u(x)$ and $v(x)$ are differentiable and non-differentiable functions respectively. The proximal gradient method using the above minimization model can be defined as,

$$x^{(k+1)} = prox_t\left(x^{(k)} - t\nabla u\left(x^{(k)}\right)\right) \tag{4.16}$$

where the proximal operator - $prox_t(.)$ is defined as,

$$prox_t(z) = \min_y \left\{ v(y) + \frac{1}{2t}\|y - z\|^2 \right\} \tag{4.17}$$

$$z = x^{(k)} - t\nabla u\left(x^{(k)}\right)$$

The derivation of this method can be found in [65], where they used the concept of Majorizer- Minimzation algorithm (MM algorithm) along with the Quadratic approximation to model the differentiable $u(x)$. Therefore, proximal gradient method can also be interpreted as an instance of the Majorizer-Minimization algorithm. A large class of algorithms that includes gradient method, Newton's method and Expectation-Maximization algorithm are some of the other special cases of Majorizer-Minimization algorithm.

In our problem, due to the existence of both differentiable and non-differentiable components in the cost function, we employ the Majorizer-Minimization scheme in our optimization model (this development will be explained in Section 4.3.3). Majorizer-Minimization algorithm uses a surrogate function that minimizes (or maximizes for a Minorize-Maximization model) the objective function, where the surrogate function will drive the objective function to a local minimum at each iteration.

Let $L(x)$ be the objective function, and $M(x)$ be its surrogate function. The surrogate function should include the following properties:

a. $M\left(x,x\right)=L\left(x\right)$ for every $x\in E$

b. $M\left(x,y\right)\geq L\left(x\right)$ for every $x,y\in E$

Let $y=x_{k-1}$, where $x_k\in\min_x(M(x,x_{k-1}))$. This implies that $M\left(x_k,x_{k-1}\right)\leq M\left(x_{k-1},x_{k-1}\right)$. Therefore, considering all of the above properties, we can write the following conclusion:

$$L\left(x_k\right)\leq M\left(x_k,x_{k-1}\right)\leq M\left(x_{k-1},x_{k-1}\right)=L\left(x_{k-1}\right)$$

Therefore,

$$L\left(x_k\right)\leq L\left(x_{k-1}\right)$$

Hence, this can be used as a minimization scheme for $L(x)$.

4.3.2 Block-Coordinate Descent (BCD) Method

BCD algorithm is based on the idea that an n-dimensional problem can be decomposed into p sub-problems, and the objective function is optimized over one such segment/block at each sub-iteration, while keeping all the other segments/blocks fixed. Even though the BCD algorithm finds the global optimum for all differentiable functions optimized by each group while keeping the other groups fixed, it does not however realize the global optimum for all non-differentiable functions. The condition to be satisfied for a non-differentiable function to attain the global optimum using BCD is that

it should be separable [64, 63]. The separability for the case of addition can be defined as follows:

Suppose there is a function F with p variables $x_1, x_2, \ldots x_p$. We say that F is additively seperable if there exist functions $f_1, f_2, \ldots f_p$ such that,

$$F\left(x_1, x_2, \ldots, x_n\right) = f_1\left(x_1\right) + f_2\left(x_2\right) + \ldots + f_n\left(x_n\right)$$

Let us consider the overall cost function used for the SGSL0 algorithm (4.8). We can see that the quadratic fidelity term and the exponential "l_0 norm" penalty operator are both differentiable. Even though the "l_2 norm" penalty is non-differentiable, it is separable among the groups l. Therefore, BCD algorithm can be used to find the global minimum of the overall cost function, as long as the penalty functions are convex and the optimization is performed group/block-wise. The proof for the global convergence using the BCD method for a function with differentiable and non-differentiable (yet separable) components can be found in [63].

4.3.3 Optimization Algorithm

Let us re-write the overall cost function for our problem (using equation (4.7)):

$$L_{SGSL0}\left(x\right) = \left[\frac{\lambda_0}{2}\left\|y - \sum_{l=1}^{p} A^l x^l\right\|_2^2 + \lambda_1 \sum_{l=1}^{p}\left\|x^l\right\|_2 + \lambda_2 \sum_{l=1}^{p}\left[n_l - \sum_{i=1}^{n_l} e^{\frac{-(x_i^l)^2}{2\sigma^2}}\right]\right] \tag{4.18}$$

Using the above criteria on separability we now optimize the above cost function using the BCD algorithm. It is important to note that the novel global optimality criterion (mentioned in (4.14)) is based on this ability to optimize for each block, while keeping the other blocks fixed.

We now decompose the overall cost function into the following sub-problems:

$$L_{SGSL0}\left(x^{l}\right)=\left[\frac{\lambda_{0}}{2}\left\|y^{-l}-A^{l}x^{l}\right\|_{2}^{2}+\lambda_{1}\left\|x^{l}\right\|_{2}+\lambda_{2}\left[n_{l}-\sum_{i=1}^{n_{l}}e^{\frac{-(x_{i}^{l})^{2}}{2\sigma^{2}}}\right]\right] \tag{4.19}$$

The above minimization problem is solved for each group l, while keeping all the other groups fixed. Here, y^{-l} is the segment of y that does not contain $A^{l}x^{l}$ and it is defined as:

$$y^{-l}=y-\sum_{k\neq l}^{p}A^{k}x^{k}$$

It is important to note that the direct application of the proximal gradient method to solve the cost function in (4.19) forces us to solve another non-linear equation, increasing computational complexity. This is caused by the exponential ("l_0 - norm" penalty) term in the cost function. Therefore, we propose to conduct the optimization by locally approximating the exponential "l_0 - norm" penalty by a quadratic function (as previously discussed in Section 4.1).

Now let us consider the minimization of $L_{SGSL0}(x^l)$ with respect to x^l. From (4.19), we can write the gradient of L_{SGSL0} with respect to x^l as:

$$\frac{\partial L_{SGSL0}\left(x^l\right)}{\partial x^l} = -\lambda_0\left(A^l\right)^T\left(y^{-l} - A^l x^l\right) + \frac{\lambda_2 W\left(x^l\right)x^l}{\sigma^2} + \lambda_1 S = \vec{0} \tag{4.20}$$

where, $S = \begin{cases} \dfrac{x^l}{\left\|x^l\right\|_2} & x^l \neq \vec{0} \\ \left\{ Z : \left\|Z\right\|_2 \leq 1 \right\} & for\ x^l = \vec{0} \end{cases}$ and $Z \in \mathbb{R}^{n_l}$ can be any vector.

Again, as we discussed before, the reason for this kind of a behavior for the gradient of $\left\|x^l\right\|_2$ is its discontinuity at $x^l = \vec{0}$. We use this as a thresholding function during our optimization process as follows:

When $x^l = \vec{0}$, from (4.20),

$$-\lambda_0\left(A^l\right)^T y^{-l} + \lambda_1 Z = \vec{0}$$

$$Z = \frac{\lambda_0}{\lambda_1}\left(A^l\right)^T y^{-l}$$

Taking the l_2 - norm from both sides of the equation,

$$\left\|Z\right\|_2 = \frac{\lambda_0}{\lambda_1}\left\|\left(A^l\right)^T y^{-l}\right\|_2 \leq 1 \tag{4.21}$$

Therefore, if $\lambda_1 \geq \lambda_0\left\|\left(A^l\right)^T y^{-l}\right\|_2$, the solution is $x^l = \vec{0}$,

For the case where $x^l \neq \vec{0}$, let's define $d_1\left(x^l\right)$, $d_2\left(x^l\right)$ and $c\left(x^l\right)$ as follows:

$$d_1\left(x^l\right) = \frac{\lambda_0}{2}\left\|y^{-l} - A^l x^l\right\|_2^2 \tag{4.22}$$

$$d_2\left(x^l\right) = \lambda_2\left[n_l - \sum_{i=1}^{n_l} e^{\frac{-(x_i^l)^2}{2\sigma^2}}\right] \tag{4.23}$$

$$c\left(x^l\right) = \left\|x^l\right\|_2 \tag{4.24}$$

$$L_{SGSL0}\left(x^l\right) = d_1\left(x^l\right) + \lambda_1 c\left(x^l\right) + d_2\left(x^l\right) \tag{4.25}$$

Here, $d_1(x^l)$ represents the convex and differentiable fidelity component, $d_2(x^l)$ represents the convex "l_0 -norm" penalty term and $c(x^l)$ represents the convex non-differentiable component, which promotes group sparsity.

Proximal gradient method will linearize $d_1(x^l)$ and $d_2(x^l)$ around it's current iterate (current point x_0) using the quadratic (Q) approximation models as follows:

$$Q_1\left(x^l, x_0\right) = d_1\left(x_0\right) + \nabla d_1^T\left(x_0\right)\left(x^l - x_0\right) + \frac{L_1}{2}\left\|x^l - x_0\right\|_2^2 \tag{4.26}$$

$$Q_2\left(x^l, x_0\right) = d_2\left(x_0\right) + \nabla d_2^T\left(x_0\right)\left(x^l - x_0\right) + \frac{L_2}{2}\left\|x^l - x_0\right\|_2^2 \tag{4.27}$$

Using the concept of Majorizer-Minimization, $Q_1(x^l, x_0) + Q_2(x^l, x_0) + \lambda_1 c(x^l)$ acts as a surrogate function to the cost function $L_{SGSL0}(x^l)$. A line-search method is used to find the constants L_1 and L_2 such that it sufficiently satisfies the following conditions:

$$Q_1\left(x^l, x_0\right) \geq d_1\left(x^l\right) \tag{4.28}$$

$$Q_2\left(x^l, x_0\right) \geq d_2\left(x^l\right) \tag{4.29}$$

Let the surrogate function be $M\left(x^l, x_0\right)$:

$$M\left(x^l, x_0\right) = Q_1\left(x^l, x_0\right) + Q_2\left(x^l, x_0\right) + \lambda_1 c\left(x^l\right) \geq L_{SGSL0}\left(x^l\right) \tag{4.30}$$

Now if we let $x_0 = x^{l,(k-1)}$ and $x^l = x^{l,(k)}$, which are the current and next iterates for x^l respectively, we can say,

$$x^{l,(k)} = \arg\min_x M\left(x, x^{l,(k-1)}\right) \tag{4.31}$$

Minimization of $Q_1(x^{l,(k)}, x^{l,(k-1)})$ and $Q_2(x^{l,(k)}, x^{l,(k-1)})$ with respect to x^l can be reduced to the following:

$$Q_1\left(x^{l,(k)}, x^{l,(k-1)}\right) = \frac{L_1}{2}\left\| x^{l,(k)} - x^{l,(k-1)} + \frac{\nabla d_1\left(x^{l,(k-1)}\right)}{L_1}\right\|_2^2 + d_1\left(x^{l,(k-1)}\right) - \frac{\nabla d_1^2\left(x^{l,(k-1)}\right)}{2L_1} \tag{4.32}$$

$$Q_2\left(x^{l,(k)},x^{l,(k-1)}\right)=\frac{L_2}{2}\left\|x^{l,(k)}-x^{l,(k-1)}+\frac{\nabla d_2\left(x^{l,(k-1)}\right)}{L_2}\right\|_2^2+ \tag{4.33}$$
$$d_2\left(x^{l,(k-1)}\right)-\frac{\nabla d_2^2\left(x^{l,(k-1)}\right)}{2L_2}$$

After removing constants we get,

$$\min_{x^{l,(k)}} M\left(x^{l,(k)},x^{l,(k-1)}\right)=\min_{x^{l,(k)}}\left[\begin{array}{l}\frac{L_1}{2}\left\|x^{l,(k)}-x^{l,(k-1)}+\frac{\nabla d_1\left(x^{l,(k-1)}\right)}{L_1}\right\|_2^2\\ \quad+\frac{L_2}{2}\left\|x^{l,(k)}-x^{l,(k-1)}+\frac{\nabla d_2\left(x^{l,(k-1)}\right)}{L_2}\right\|_2^2\\ \qquad+\lambda_1 c\left(x^{l,(k)}\right)\end{array}\right] \tag{4.34}$$

Since $c\left(x^{l,(k)}\right)=\left\|x^{l,(k)}\right\|_2$, we have

$$\min_{x^{l,(k)}} M\left(x^{l,(k)},x^{l,(k-1)}\right)=\min_{x^{l,(k)}}\left[\begin{array}{l}\frac{L_1}{2}\left\|x^{l,(k)}-x^{l,(k-1)}+\frac{\nabla d_1\left(x^{l,(k-1)}\right)}{L_1}\right\|_2^2\\ \quad+\frac{L_2}{2}\left\|x^{l,(k)}-x^{l,(k-1)}+\frac{\nabla d_2\left(x^{l,(k-1)}\right)}{L_2}\right\|_2^2\\ \qquad+\lambda_1\left\|x^{l,(k)}\right\|_2\end{array}\right] \tag{4.35}$$

Let us now take the gradient of $M\left(x^{l,(k)},x^{l,(k-1)}\right)$ with respect to x^l,

$$\frac{\partial M\left(x^{l},x^{l,(k-1)}\right)}{\partial x^{l,(k)}}=(L_1+L_2)x^{l,(k)}-R\left(x^{l,(k-1)}\right)+\lambda_1 S=\vec{0} \tag{4.36}$$

where,

$$R\left(x^{l,(k-1)}\right)=\left(L_1+L_2\right)x^{l,(k-1)}-\nabla d_1\left(x^{l,(k-1)}\right)-\nabla d_2\left(x^{l,(k-1)}\right) \tag{4.37}$$

Let x^{l} to be the next iterate $x^{l,(k)}$ in equation (4.36) and let us consider the case for $x^{l,(k)}=\vec{0}$,

$$\lambda_1 Z=R\left(x^{l,(k-1)}\right)$$

But we know that $\|Z\|_2\le 1$,

$$\|Z\|_2=\frac{\left\|R\left(x^{l,(k-1)}\right)\right\|_2}{\lambda_1}\le 1$$

Therefore, if,

$$\lambda_1\ge\left\|R\left(x^{l,(k-1)}\right)\right\|_2 \tag{4.38}$$

then we get a solution $x^{l,(k)}=\vec{0}$.

Now let us consider the case for $x^{l,(k)}\neq\vec{0}$,

$$(L_1+L_2)x^{l,(k)}-R\left(x^{l,(k-1)}\right)+\lambda_1\frac{x^{l,(k)}}{\left\|x^{l,(k)}\right\|_2}=\vec{0}$$

$$\left(L_1 + L_2 + \frac{\lambda_1}{\left\| x^{l,(k)} \right\|_2} \right) x^{l,(k)} = R\left(x^{l,(k-1)} \right) \tag{4.39}$$

Taking the l_2 - norm from both sides,

$$\left\| x^{l,(k)} \right\|_2 \left(L_1 + L_2 + \frac{\lambda_1}{\left\| x^{l,(k)} \right\|_2} \right) = \left\| R\left(x^{l,(k-1)} \right) \right\|_2$$

$$\left\| x^{l,(k)} \right\|_2 = \frac{\left\| R\left(x^{l,(k-1)} \right) \right\|_2 - \lambda_1}{\left(L_1 + L_2 \right)}$$

In other words, using the knowledge from equation (4.38), we can define

$\left(X \right)_+ = \max\left(X, 0 \right)$ and incorporate it in our optimization model as follows,

$$\left\| x^{l,(k)} \right\|_2 = \left(\frac{\left\| R\left(x^{l,(k-1)} \right) \right\|_2 - \lambda_1}{\left(L_1 + L_2 \right)} \right)_+ \tag{4.40}$$

The above equation combines the two cases for $x^l : x^l = \vec{0}$ and $x^l \neq \vec{0}$.

Equation (4.39) can be re-written as,

$$\left(L_1 + L_2 + \frac{\lambda_1 \left(L_1 + L_2 \right)}{\left\| R\left(x^{l,(k-1)} \right) \right\|_2 - \lambda_1} \right) x^{l,(k)} = R\left(x^{l,(k-1)} \right)$$

Therefore,

$$x^{l,(k)} = \left(\frac{\left\| R\left(x^{l,(k-1)} \right) \right\|_2 - \lambda_1}{\left\| R\left(x^{l,(k-1)} \right) \right\|_2 \left(L_1 + L_2 \right)} \right) R\left(x^{l,(k-1)} \right)$$

The final iterative update equation for both $x^l = \overrightarrow{0}$ and $x^l \neq \overrightarrow{0}$ can be combined together as,

$$x^{l,(k)} = \left(\frac{1}{L_1 + L_2} - \frac{\lambda_1}{\left\| R\left(x^{l,(k-1)}\right) \right\|_2 \left(L_1 + L_2\right)} \right)_+ R\left(x^{l,(k-1)}\right) \tag{4.41}$$

where, using Equation (4.22) and Equation (4.23),

$$\nabla d_1\left(x^{l,(k-1)}\right) = -\lambda_0 \left(A^l\right)^T \left(y^{-l} - A^l x^{l,(k-1)}\right) \tag{4.42}$$

$$\nabla d_2\left(x^{l,(k-1)}\right) = \frac{\lambda_2 W\left(x^{l,(k-1)}\right) x^{l,(k-1)}}{\sigma^2} \tag{4.43}$$

The value for $\sigma_{\min}$ is chosen using a-priori knowledge about the strength of the signal to be reconstructed and using experimental results. From both experimental results and using the criterion given in [46], we choose 0.01 as the most suitable value for $\sigma_{\min}$. It is important to note that, for a case where the global criterion is initially not satisfied for $\sigma_{\min}$, we utilize a sequence of σ's in a descending order. This careful selection is expected to avoid any local minima and end up with the global minimum.

Algorithm 2 – SGSL0 Algorithm overview

1. Input : $\lambda_1, \lambda_2, A \in \mathbb{R}^{m \times n}, y \in \mathbb{R}^m, n_l, p, q_l$

2. Initialization : Using Minimum Norm Solution

3. $x_0 = (A^T A)^{-1} A^T y$

4. *for* $k = 1$ *to* $k_{\max}$ *do*

5. *for* $l = 1$ *to* p *do*

6. *if* $\lambda_1 \geq \left\| R(x^{l,(k-1)}) \right\|_2$ *then*

7. $x^{l,(k)} = \vec{0}$

8. *else if* (4.14) satisfied for $\sigma_{\min}$ *then*

9. Find L_1, L_2 and execute (4.41) to find $x^{l,(k)}$

10. *else*

11. *for* $\sigma_{\max}$ *to* $\sigma_{\min}$ *do*

12. Find L_1, L_2 and execute (4.41) to find $x^{l,(k)}$

13. *end for*

14. *end if*

15. *end for*

16. *end for*

4.4 Non-Stationary Signal Reconstruction

In the earlier works of this Chapter, we assume the active brain sources to be temporally smooth. This assumption helps us to formulate the ill-posed problem as a vector

reconstruction problem as opposed to a matrix reconstruction problem. In other words, we recover the unknown source vector $x \in \mathbb{R}^n$ instead of recovering a spatio-temporal source matrix $X \in \mathbb{R}^{n \times t}$, where t is the number of time samples.

In some cases, we can observe the brain sources to depict non-stationary and transient-like behavior [21] during the time interval of interest. This section is devoted to discuss how mathematically we can include those physiologically motivated priors in our novel source localization algorithm.

We first define the underdetermined problem for the spatio-temporal case as follows:

$$Y = AX + E \tag{4.44}$$

$$\min_X \left[\|Y - AX\|_2^2 + \lambda \Omega(X) \right] \tag{4.45}$$

where $Y \in \mathbb{R}^{m \times t}, A \in \mathbb{R}^{m \times n}, X \in \mathbb{R}^{n \times t}, \mathrm{E} \in \mathbb{R}^{m \times t}, \Omega(X)$ are the Sensor matrix, Gain matrix, Source Matrix, Noise Matrix and the prior inducing function, respectively. As we discussed before, since SGSL0 captures both group-wise and feature level sparsity, we can extend it to capture both stationary and non-stationary activations while recovering the signal matrix X.

The four possible cases of sparsity for a given matrix X can be categorized as: spatially group-wise sparse activations, spatially focal/ feature level sparse activations,

temporally group-wise (stationary) sparse activations and temporally feature level sparse activations. These four cases are depicted in Figure 4.1.

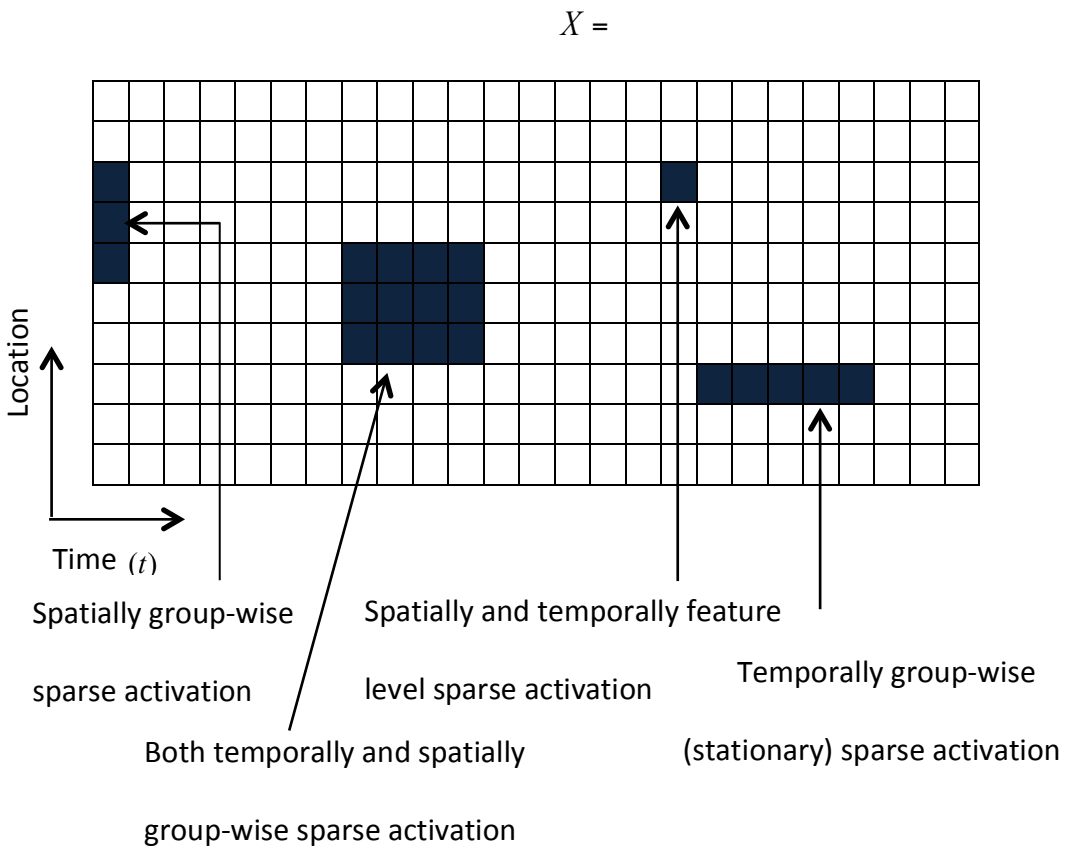


Figure 4.1: Spatio-Temporal source matrix

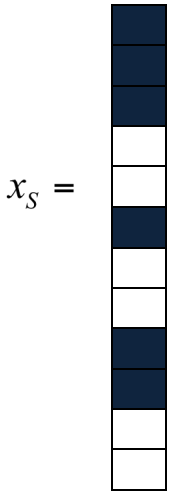


Figure 4.2: Super vector of the source matrix

By observing the columns in the above X matrix, we notice that both spatial and temporal group-wise and feature-level sparsity can be addressed using the SGSL0 optimiation model given in (4.7). In order to apply the SGSL0 algorthm, we convert the X matrix into a super vector $x_s \in \mathbb{R}^{nt\times 1}$ (Figure 4.2), which is formed by stacking each column of X on top of each other. Similarly, we convert the sensor matrix Y into a super vector $y_s \in \mathbb{R}^{mt\times 1}$ and replace the gain matrix A by its Kronecker product with the identity matrix $I_t \in \mathbb{R}^{t\times t}$. The regression model in (4.44) can now be written as:

$$y_s = \tilde{A}x_s \tag{4.46}$$

where $\tilde{A} = A \otimes I_t$.

Thereafter, we use the a-priori knowledge about the structure of both spatial activation and temporal activation to define corresponding groups and group lengths (n_l) to recover the super vector x_s using the SGSL0 algorithm. We include the simulations of this non-stationary signal recovery case in Section 5.5.

5 SIMULATION STUDIES

In order to assess the performance of the proposed algorithms, we carried out some experiments to be presented in this chapter. The performance of a sparse reconstruction method mainly relies on two factors: the sparsity of the signal (i.e., the number of active sources) and the ratio between the number of (potential) sources against the number of sensors. We based our experiments mainly on these two factors which are also used as performance measures.

In Section 5.1 we simulate the algorithm introduced in Section 4.1 (Algorithm 1), which is for no grouping and is part of our original contribution. We refer to it as the Quadratic majorizer based SL0. We compare these results with currently used models such as LASSO (l_1- norm regularization) [$l_1$magic – [41]] and m-FOCUSS (l_p -norm $(0 < p < 1)$ [68]). In Section 5.2 we simulate Algorithm 2 developed in Chapter 4 – Sparse Group SL0. Finally, in Section 5.4, we use the same SGSL0 algorithm to solve the inverse problem of a simulated MEG problem.

5.1 Quadratic majorizer based SL0 (QSL0)

For the following experiments, the Gain matrix A is chosen as a Random Gaussian matrix with mean 0 and standard deviation 1. The signal to be reconstructed - $x*$ is chosen to have values 0, 1, 2 and 4. The locations of the non-zero entries are selected such that five entries of the same non-zero values are adjacent to each other. Such locations are selected randomly as shown in Figure 5.1(a). The observation vector y is then generated using the model $y = Ax*$.

First, as seen in Figures 5.1 and 5.2, we compare the QSL0 reconstruction with the original signal model, as we change the level of sparsity (k). We referred to the original experiments done by [46] when selecting the parameters, where QSL0 showed comparably better reconstruction for a selected set of regularization parameters. As seen from Figure 5.1 to Figure 5.2, increment in the level of sparsity deteriorates the reconstruction substantially.

Next, we compare the QSL0 method with the current standard l_1 norm method along with the FOCUSS method. We use the publicly available l_1 -magic [41] package to derive the l_1 norm based solution, and we use the publicly available [68] m-FOCUSS algorithm to derive the l_p norm $(0 < p < 1)$ based solution for two different p values. In these experiments, we use the Peak Signal-to-Noise Ratio (PSNR), defined below, to assess the quality of the reconstruction.

Given $x \in \mathbb{R}^n$ and its estimation $v \in \mathbb{R}^n$,

$$PSNR(v) = 20\log_{10}\left(\frac{x_{max}}{rmse}\right)$$

where, $rmse = \sqrt{\sum_{i=1}^{n} \frac{(v_i - x_i)^2}{n}}$ and $x_{max} = \max_{i=1\ldots n} |x_i|$

As seen from Figures 5.3 (a) – 5.3 (c), the only time QSL0 is out performed by l_1- norm is when m < 380 and k =120. For all the other cases, QSL0 appears to be the superior model for reconstruction.

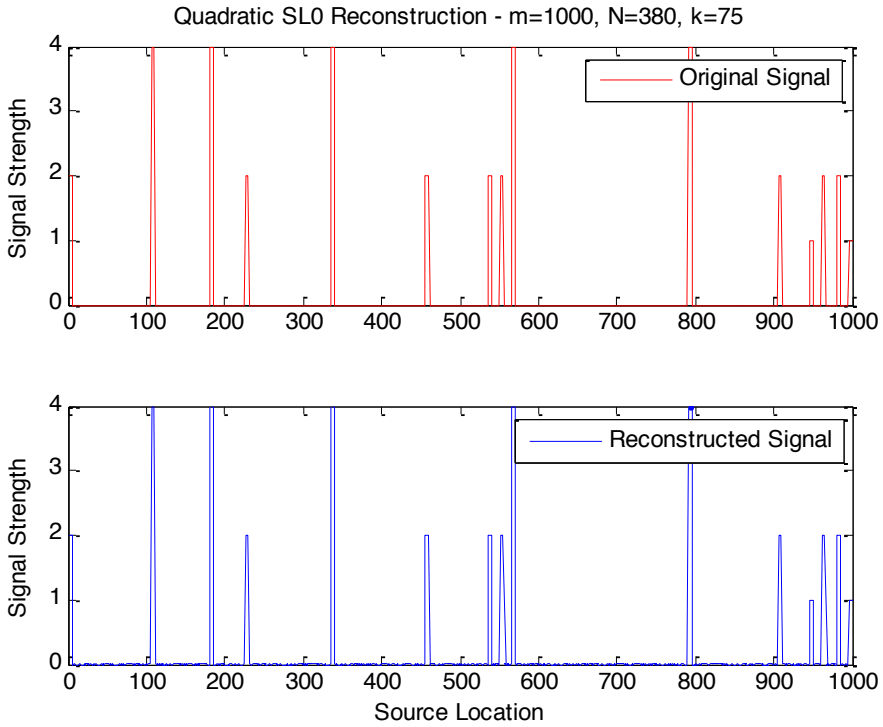


Figure 5.1 (a): Perfect Reconstruction using QSL0, $k = 75$

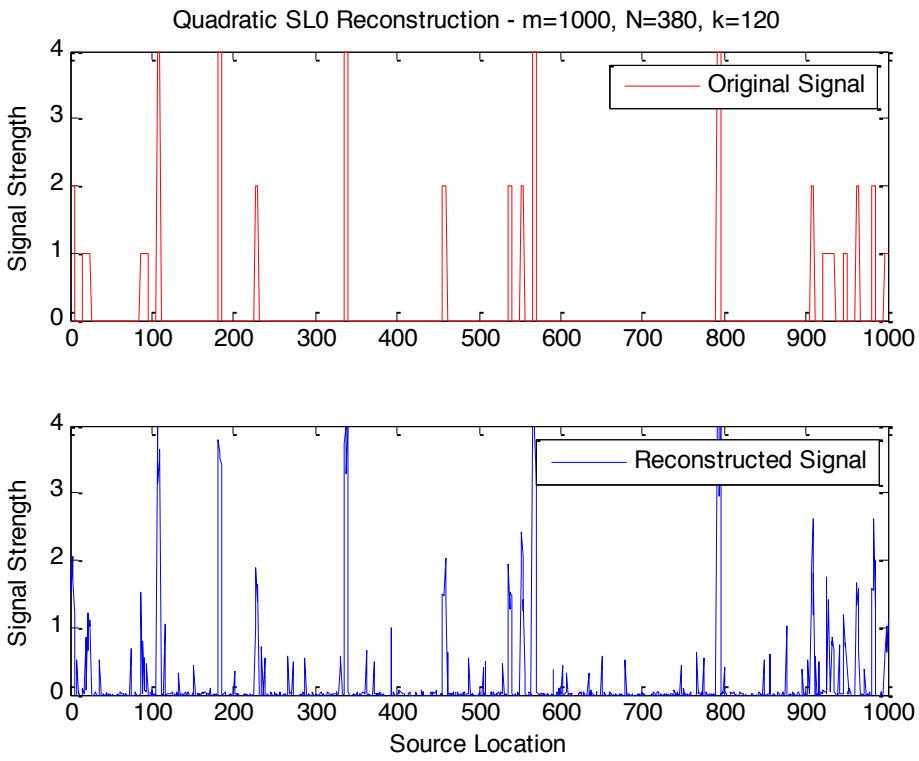


Figure 5.2: Deteriorated Reconstruction using QSL0, $k = 120$

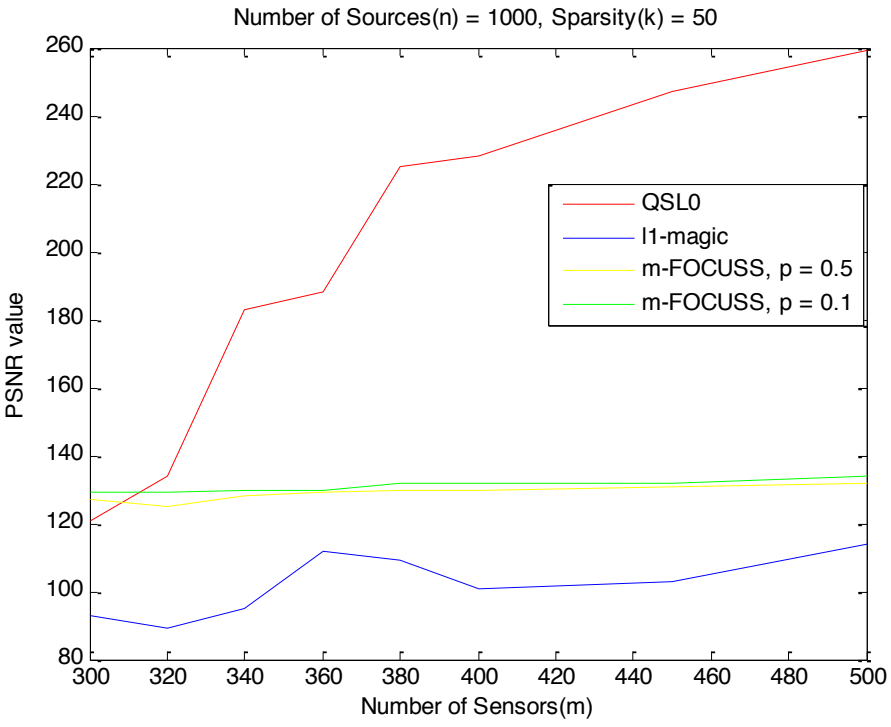


Figure 5.3 (a) PSNR Value comparisons with $k = 50$

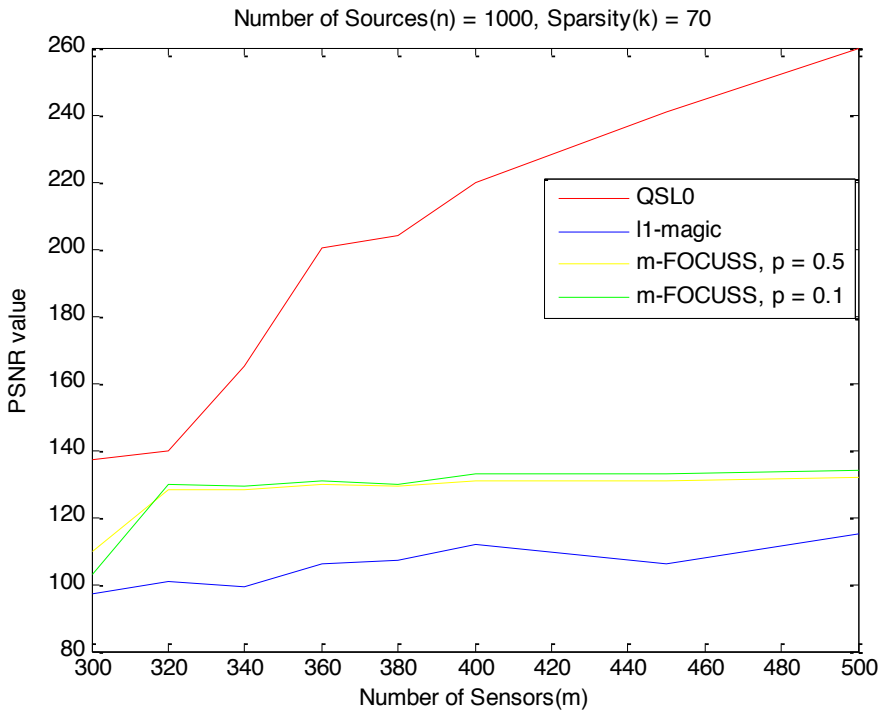


Figure 5.3 (b) PSNR Value comparisons with $k = 70$

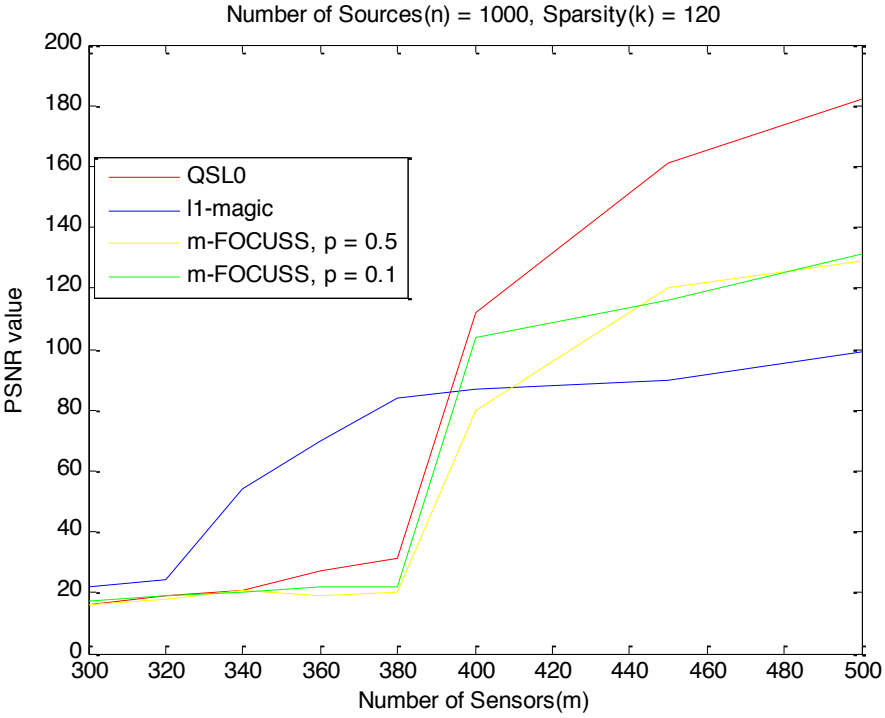


Figure 5.3 (c) PSNR Value comparisons with $k = 120$

5.2 Sparse Group SL0 (SGSL0)

During the following experiments, the Gain matrix A is chosen as a Random Gaussian matrix with mean 0 and standard deviation 1. We consider the original signal x^* being a k -sparse vector with values 1 and 0. The location of the 1's are randomly chosen such that they show group-wise sparsity and within group sparsity. The observation vector y is generated using the model $y = Ax^*$. In experiments depicted from Figures 5.4 to 5.6, we initially increased the number of sensors, while keeping the sparsity fixed (This can be seen from the corresponding sub-figures (a), (b) and (c) for each figure numbered 5.4 – 5.6). Then, we increased the sparsity level k , and observed the behavior of the algorithm (This can be seen from comparing the corresponding sub-figures from 5.4 - 5.6). The group size selected for these experiments is fixed at 100 sources per each group.

From the following figures, it is obvious that the algorithm gives better solutions when the sparsity is low, and the number of sensors is high.

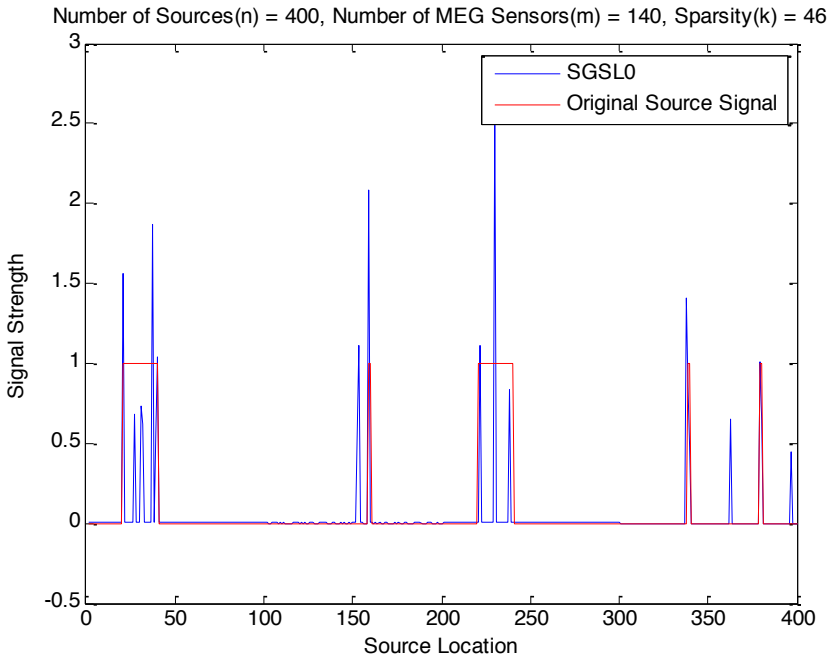


Figure 5.4 (a) SGSL0 with sensors = 140, sparsity level (k) = 46

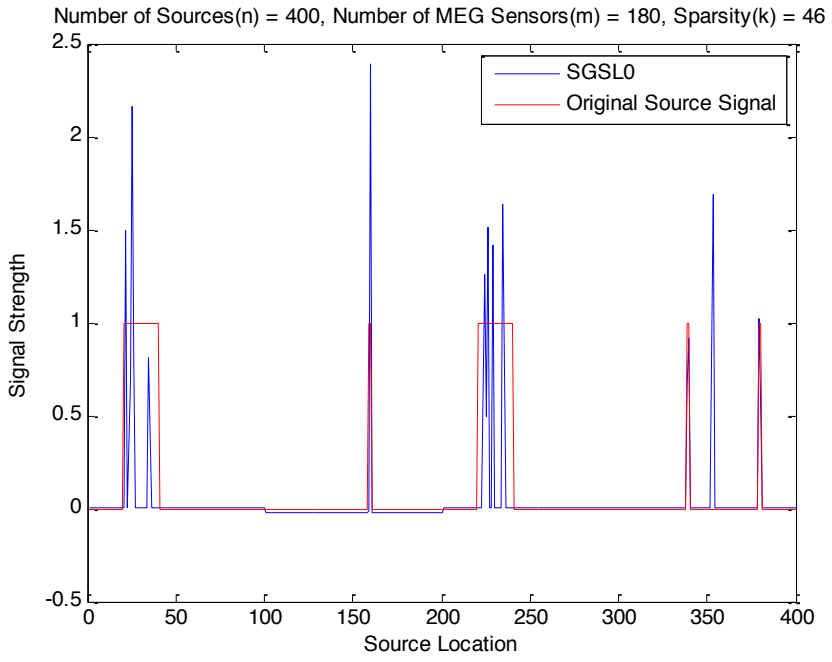


Figure 5.4 (b) SGSL0 with sensors = 180, sparsity level (k) = 46

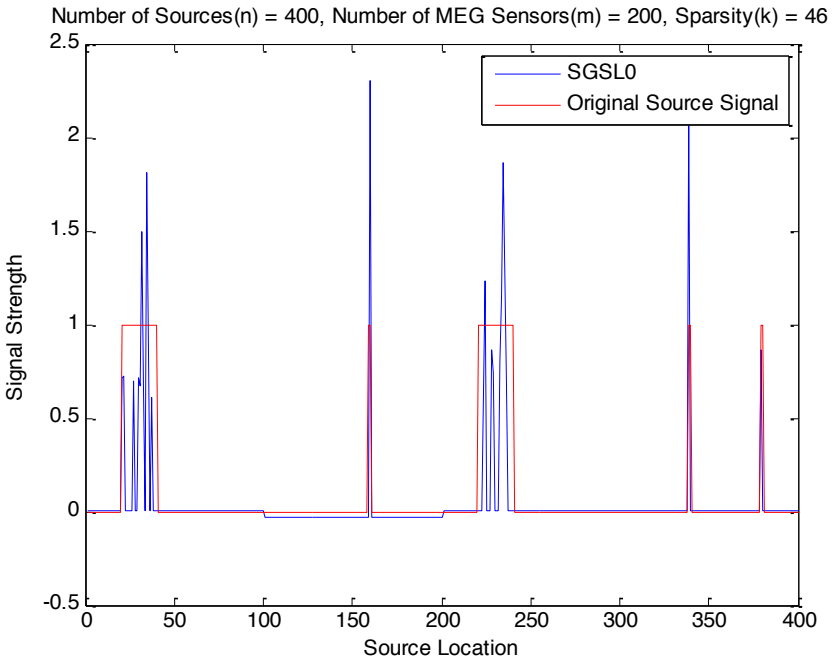


Figure 5.4 (c) SGSL0 with sensors = 200, sparsity level (k) = 46

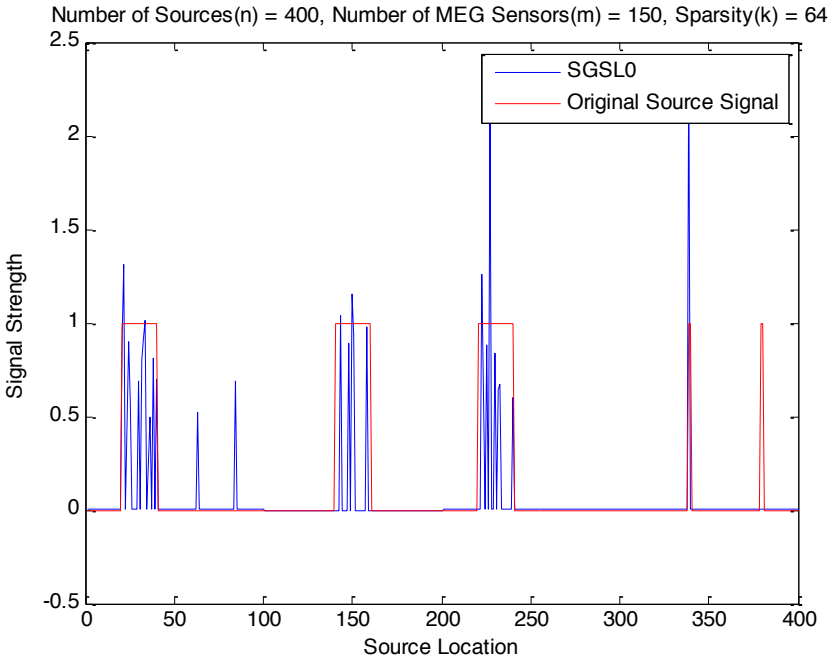


Figure 5.5 (a) SGSL0 with sensors = 150, sparsity level (k) = 64

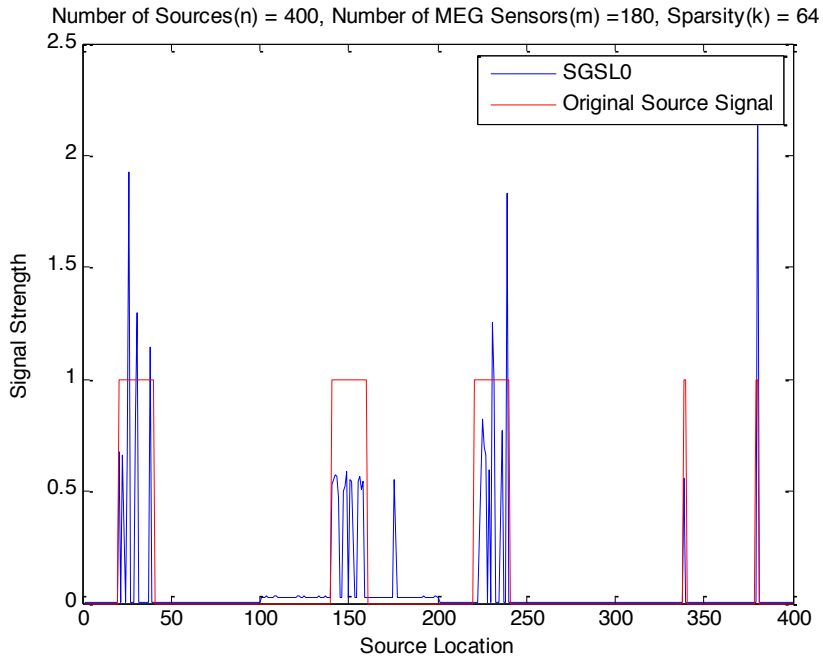


Figure 5.5 (b) SGSL0 with sensors = 180, sparsity level (k) = 64

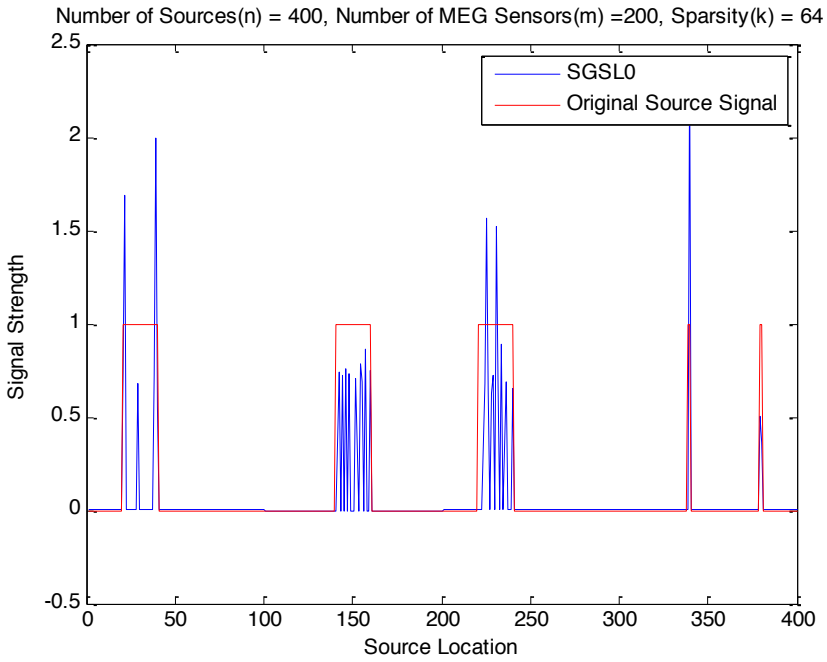


Figure 5.5 (c) SGSL0 with sensors = 200, sparsity level (k) = 64

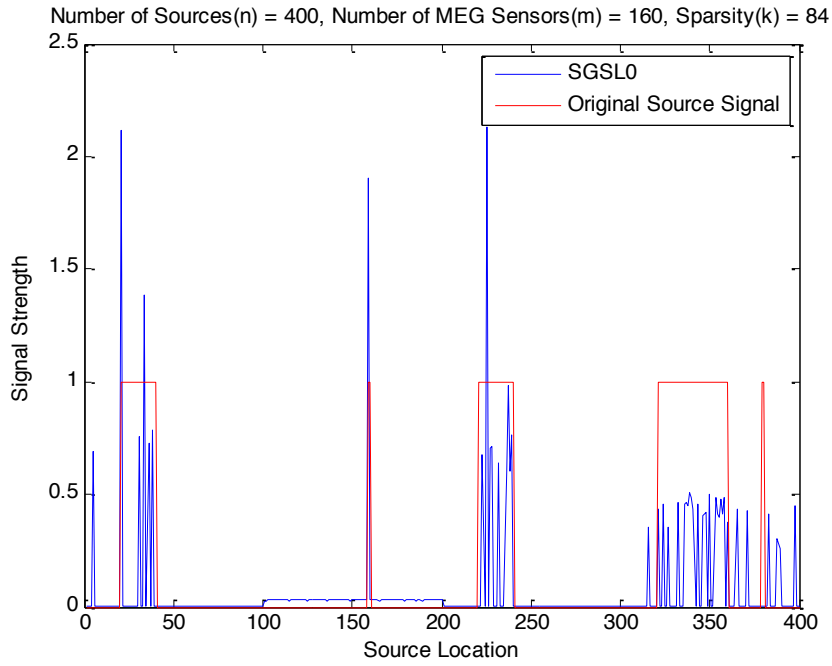


Figure 5.6 (a) SGSL0 with sensors = 160, sparsity level (k) = 84

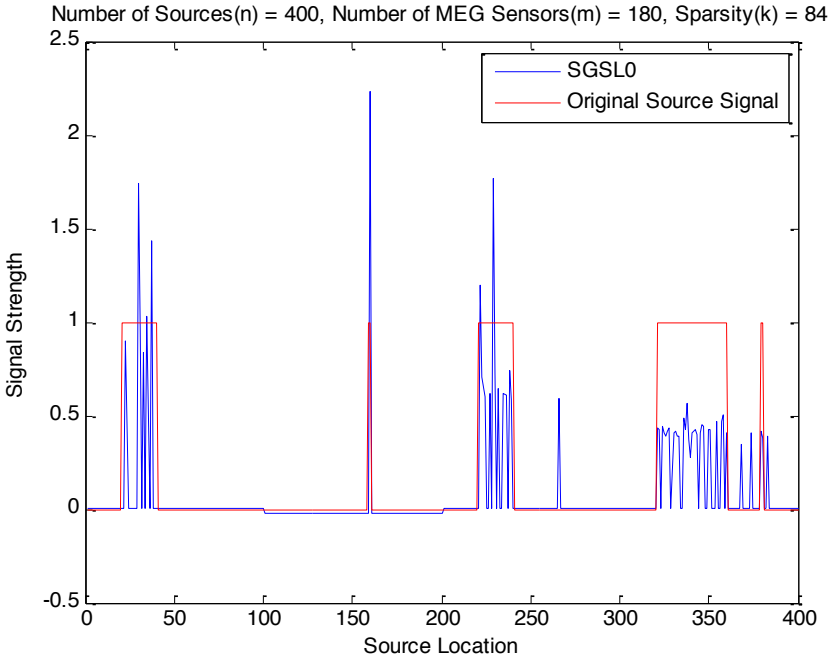


Figure 5.6 (b) SGSL0 with sensors = 180, sparsity level (k) = 84

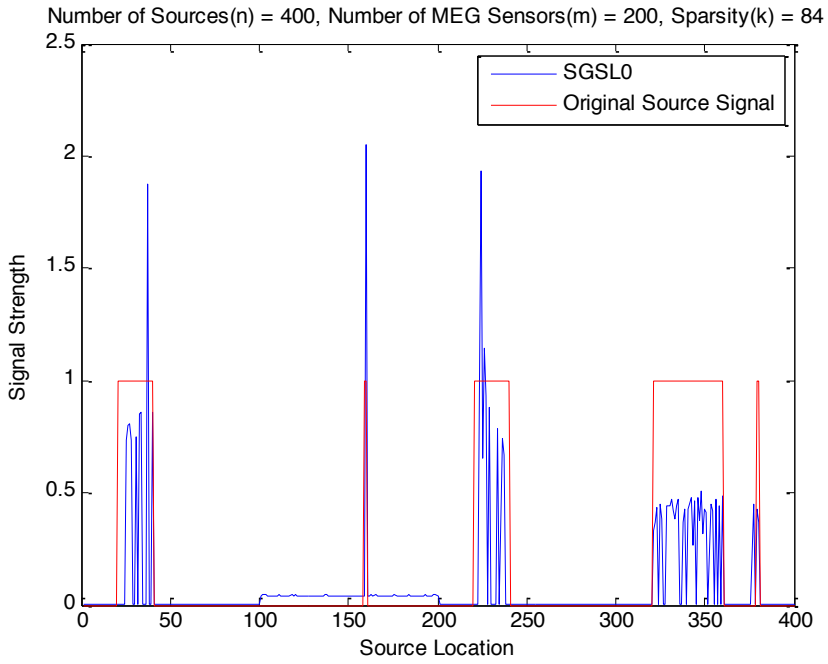


Figure 5.6 (c) SGSL0 with sensors = 200, sparsity level (k) = 84

5.2.1 Comparison between SGSL0 and Sparse Group Lasso (SGL)

We compare the performance of SGSL0 method with the currently used SGL method [12]. For this comparison we use publicly available Matlab package SLEP 4.1 [88]. To assess the quality of reconstruction, we use the following measures: the number of groups misclassified and the number of features misclassified. A misclassified group is defined to have at least one non-zero coefficient in a group whose estimated coefficients are all zero, or vice versa. Similarly, a misclassified feature is an individual coefficient estimated to be non-zero when the true coefficient is zero, or vice versa.

The gain matrix A is chosen as a Random Gaussian matrix with mean 0 and standard deviation 1. We perform two experiments using two different numbers of sensors (n): n = 800 and n = 1000. Then we observe the performance w.r.t different number of observations, i.e., different number of sensors (m). We consider the original signal x^* being a k -sparse vector with values 2, 1 and 0. The cardinality of the support of vector x^* is k = 170. The non-zero values are chosen such that they exhibit both group-wise sparsity and feature level sparsity. A fully activated group would have 40 non-zero values together, while other non-zero values will be more spread-out. The locations of the nonzero values are randomly chosen such that they show group-wise and within group sparsity. The observation vector y is generated using the model in (2.2), and Guassian noise with standard deviation 4.0 is added to each observation. As for the descending σ parameters (when the (4.14) condition is not satisfied) we use σ = [5, 1, 0.7, 0.5, 0.3, 0.1]. We use λ_0 =1, λ_1 = 1 and λ_2 = 10 for all our experiments as they showed comparatively the best results.

Following are the results obtained after averaging 50 trials of the above experiment.

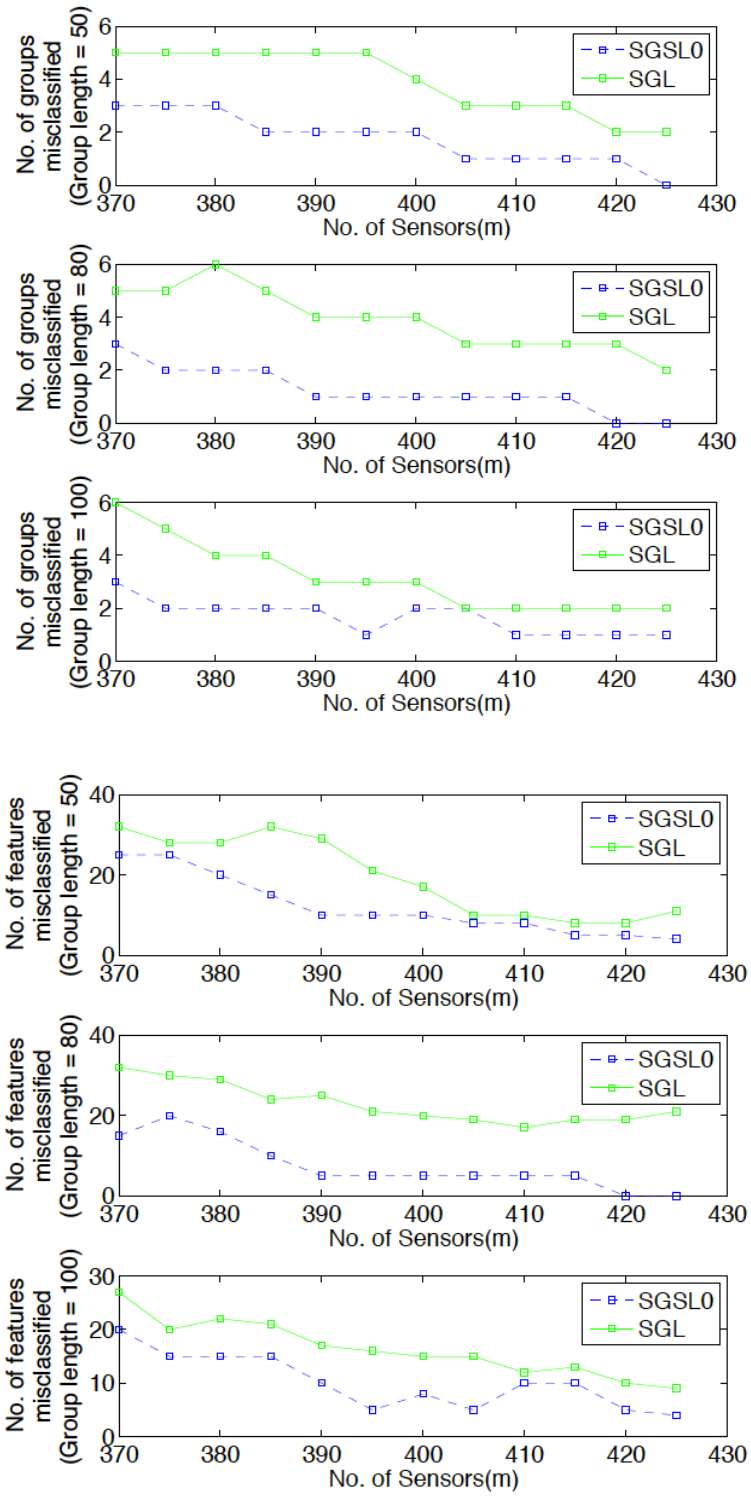


Figure 5.7: Performance Comparison when $n = 800$ for group lengths $50, 80$ and 100

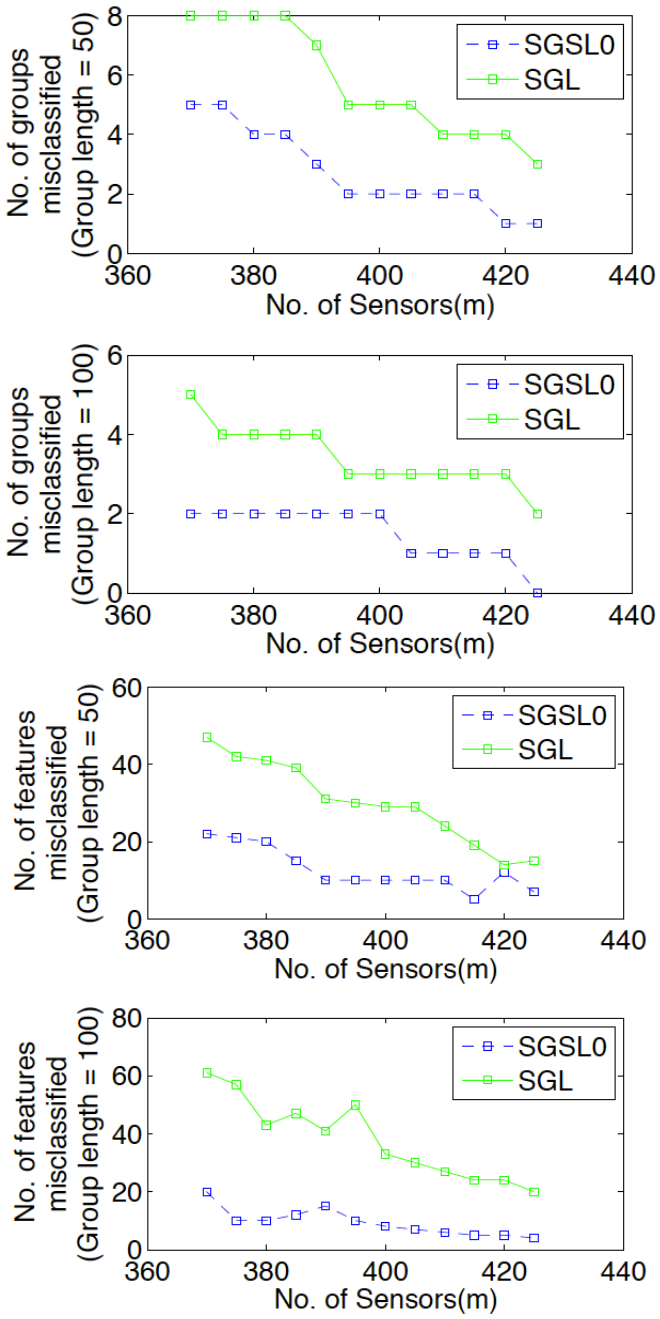


Figure 5.8: Performance Comparison when $n = 1000$ for group lengths 50 and 100

As we can see from Figures 5.7 and 5.8, the novel SGSL0 method shows superior performance compared to the SGL method.

5.3 Effect of the Regularization parameters

It is important to realize that the tuning of the regularizing parameter/s is vital to achieving a good solution. As seen in Figure 5.9, if we do not use an acceptable sequence of regularization parameters, the solution becomes deteriorated. Due to this reason, we introduce a Model Selection criterion in Chapter 7 which facilitates finding the best regularization parameters from a candidate set of parameters. In order to demonstrate the effects of the regularization parameter/s selected towards the final reconstruction, we refer the best regularization parameters chosen via trial and error as a "refined" sequence of parameters, while a set of randomly selected parameters as "unrefined" in the figure below.

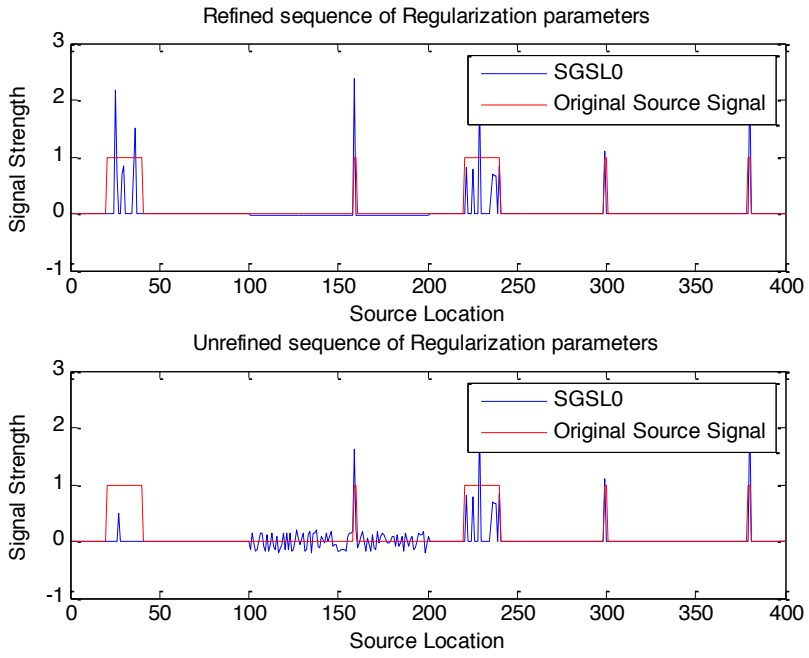


Figure 5.9 Importance of the correct selection of Regularization parameters

5.4 MEG Simulation

For MEG simulation experiments, we mainly focused on the lead-field matrix. We used the publicly available Matlab toolbox – Fieldtrip [69] in order to perform the source reconstruction using SGSL0. The source reconstruction pipeline, or in other words the inverse solution requires the following steps using fieldtrip:

1. A) Processing of Anatomical data – This involves the pre-processing of the Anatomical data, computing the volume conduction model, and computing the source model. We use the dataset provided by the fieldtrip toolbox [ftp://ftp.fcdonders.nl/pub/fieldtrip/tutorial/Subject01.zip], which has the necessary MRI data to compute the volume conduction model and the source model. We used the "singleshell" method to prepare the head model. Figure 5.8 depicts the skin of the head model and the sensors co-registered on the skin.

   B) Processing of Functional data – This involves the pre-processing of the MEG data, averaging and noise covariance estimation. We use the functions *ft_timelockanalysis* and *ft_preprocessing* provided in the Fieldtrip toolbox to do the averaging and noise covariance estimation.

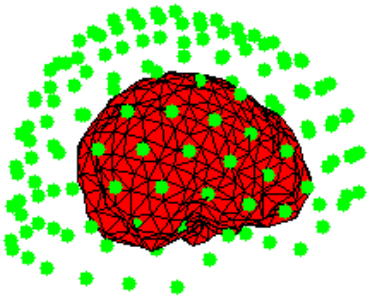


Figure 5.8: Co-registration of the MEG sensors on the head-model

2. Computation of the Forward Solution – Once the source space, the volume conduction model, and the position of the sensors are computed, the lead-field matrix can be computed using the *ft_prepare_leadfield* function. The lead-field matrix is computed as a tensor, where the dimensions are: Number of Sensors $(m)\times$Number of Sources $(n)\times$ Orientation of the Source (3). For the computation of the lead-field we use the "Boundary Element Model" [70]. It is also important to note that, we use the concept of "free orientation" [16, 71], where we find the l_2 - norm average of the 3 orientations to make it one scalar representation. This will make the inverse solution independent from the dipole orientations as we now have one single value representing the strength of the dipole signal. This will reduce the lead-field into an $m \times n$ matrix.

3. Inverse Solution and Visualization – We use the SGSL0 method to reconstruct the source model. We select the original source distribution to comprise two group-wise

activations and one focal like activation (Figure 5.9 (c)). We use the function *ft_plot_mesh* to visualize the 3-D source model as depicted in Figures 5.9 (a) and (b).

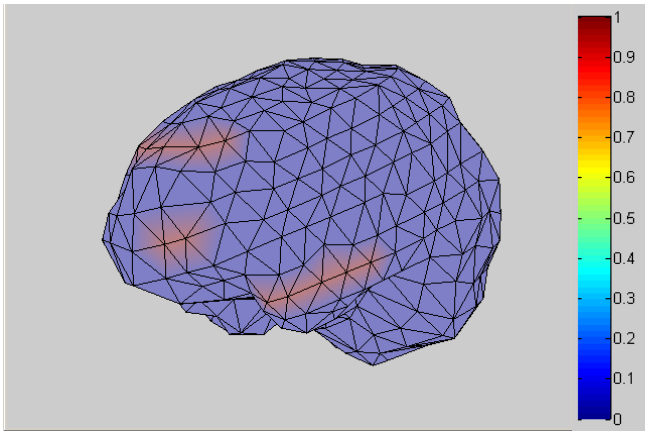


Figure 5.9 (a): Original Source Model

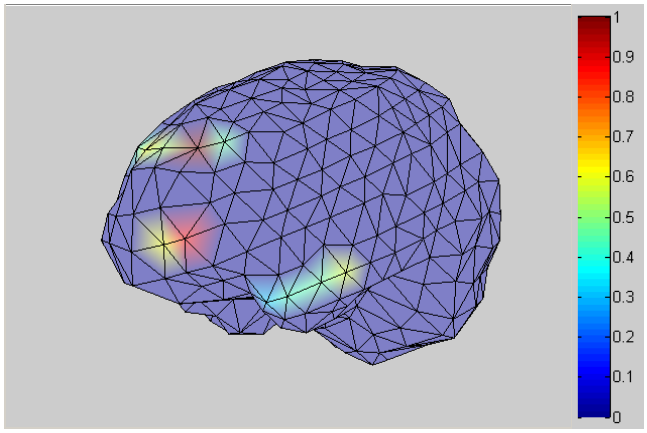


Figure 5.9 (b): Estimated Source Model

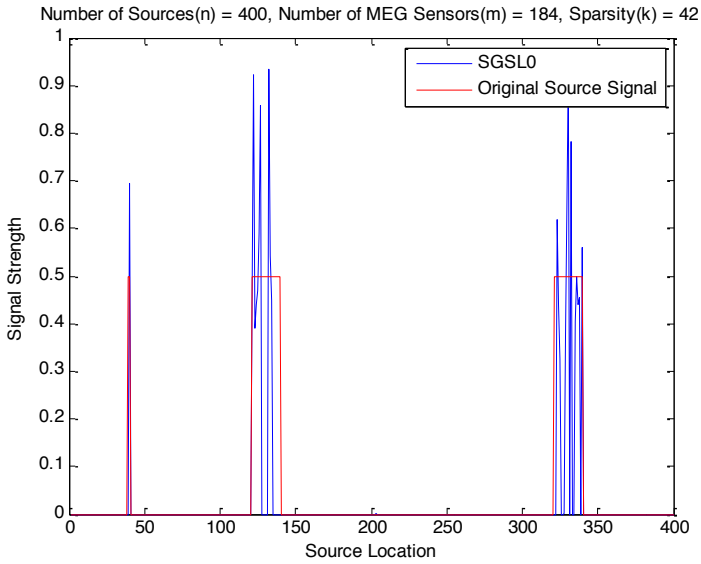


Figure 5.9 (c) : Source reconstruction using SGSL0 for a given Lead-field matrix

5.5 Non-stationary Signal Reconstruction

For this experiment, we use the mathematical construction described in Section 4.4. The dimensions of the sensor and source matrices are $Y \in \mathbb{R}^{100\times10}$ and $X \in \mathbb{R}^{250\times10}$ respectively, where the number of time samples is $t = 10$.

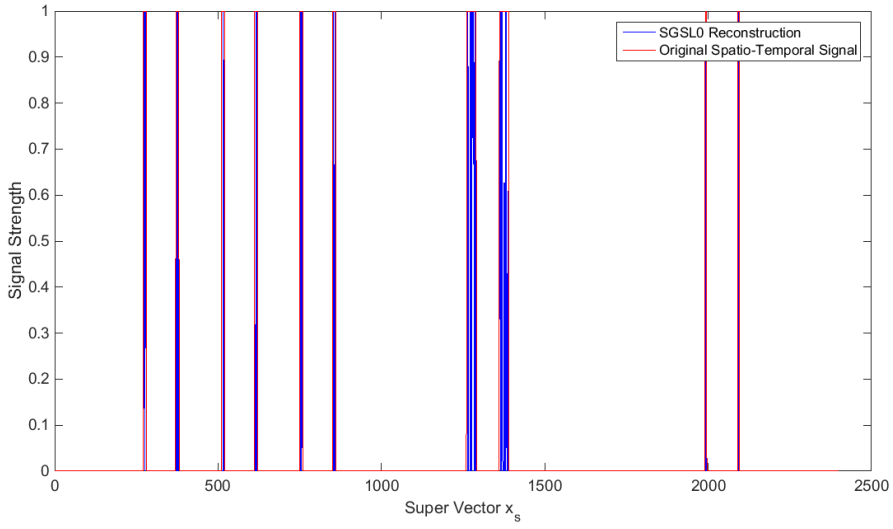


Figure 5.10: Super vector reconstruction using the SGSL0 algorithm

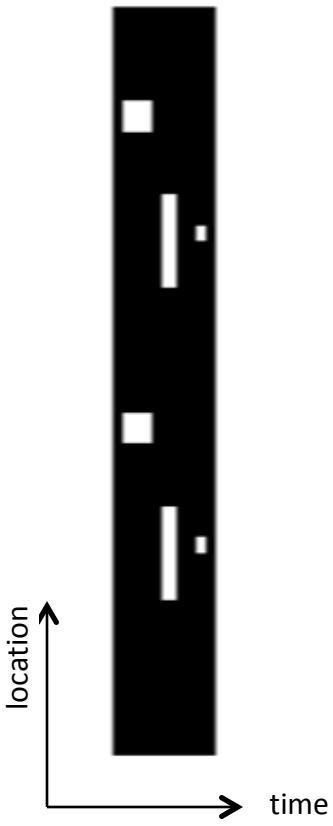


Figure 5.11(a): Original

Source Matrix - X

Figure 5.11(b): Reconstructed

Source Matrix - $\hat{X}$

The gain matrix $A \in \mathbb{R}^{100\times250}$ is chosen to be a Random Gaussian matrix with mean 0 and standard deviation 1. We design the source matrix (Figure 5.11(a)) to exhibit group-wise sparsity, focal sparsity and non-stationary behaviors. Then the sensor vector Y is obtained by using the relationship $Y = AX$. Following the matrix to super vector transition described in Section 4.4, we compute $y_s \in \mathbb{R}^{1000\times1}$ and $\tilde{A} \in \mathbb{R}^{1000\times2500}$ accordingly. Then we reconstruct the signal x_s using the SGSL0 algorithm using the same σ values described in Section 5.2.1. The resulting super vector x_s is then converted back into a matrix $\hat{X}$ and compared against the original source matrix as shown in Figure 5.11(b).

Therefore, by observing the results in Figure 5.11, we can conclude that as long as we have good a-priori knowledge about the structure of the signal, SGSL0 algorithm can be used to reconstruct both stationary and non-stationary signals.

6 GLOBAL CONVERGENCE ANALYSIS FOR SPARSE GROUP SL0

Global convergence analysis is a difficult and a challenging problem when the cost function to be optimized has many local optima. In SGSL0 algorithm, when the Global Optimality condition (4.14) is not satisfied during the initial verification for $\sigma_{\min}$, the optimization process has to be done for a sequence of σ's in a descending order. During this process, for each σ the Global Optimality condition will be checked to avoid any local minima. Unfortunately, there is no guarantee that the selected set of σ's will eventually lead to the global optimum in this scenario. Therefore, we devote this chapter to give a comprehensive theoretical foundation for this problem, and ultimately discuss on how to find the selection criterion for a set of σ's which guarantees global optimum.

In order to tackle the problem of Global Convergence Analysis of the SGSL0 method, we primarily started researching from two leads. First approach was to follow the workings in [47] as mentioned in Section 6.1 and 6.2. The second approach was to follow the works of Negabhan et al. [77], [78] to find bounds on the quantity $\|x - x^*\|_2$ for the algorithm SGSL0 as σ moves from σ_1 to $\sigma_{\min}$. One of the main challenges we faced

during the second approach was to find a corresponding regularization parameter for the SGSL0 case which satisfies the Equation (16) of Lemma 1 in [77]. In order to derive this condition, the authors in [77] used the Holder's Inequality in their proof [78]. They were able to use Holder's Inequality in their derivation since they were only interested about the l_1 norm regularizer case. But since we are using an approximated l_0 norm regularizer, which is more similar to an l_p norm regularizer with $0 < p < 1$ we would not be able to use the Holder's Inequality as it reverses itself for the case of $0 < p < 1$. Therefore, we followed the convergence analysis described in [47] (first approach mentioned above) which is less ambiguous and more relevant.

In almost all the literature pertaining to SL0 based methods [46, 87]; the selection of the sequence of σ's has been through experimental *a-priori* knowledge. In [47], Mohimani *et al*. provides a complete convergence analysis including the selection criterion of a set of σ's which will guarantee a global minimum. This criterion was based on Asymmetric Restricted Isometry Constants (ARIC's), which are hard to find in a practical sense (ARIC's depend on the gain matrix A , and when the scale of the system increases the computational complexity of finding ARIC's grows exponentially). But, they were able to show that when the gain matrix is a random Guassian matrix, the bounds for the ARIC's could be found with a high probability. We use similar arguments as described in [47] for our global convergence analysis in SGSL0 as well.

6.1 Convergence Analysis for Smoothed l_0 (SL0) method

In this section, we explain the essential theory components behind the global convergence analysis for the SL0 method mainly extracted from the work in [47]. This would set the basis for the subsequent explanation on SGSL0's global convergence. The SL0 minimization model can be described as follows:

$$\min_x \frac{1}{2}\|y - Ax\|_2^2 + \lambda\|x\|_0 \tag{6.1}$$

In this section, we approximate $\|x\|_0$ by the continuous function $n - F_\sigma(x)$ where $F_\sigma(x) = \sum_{i=1}^{n} f_\sigma(x_i)$, and focus on maximizing $F_\sigma(x)$, which is essentially the same as minimizing $n - F_\sigma(x)$. The SL0 algorithm will try to maximize $F_\sigma(x)$ using the steepest ascent method while decreasing the σ value at each outer-most iteration (refer the SGSL0 Algorithm in Section 4.3.3). The theoretical development in [47] is based on the Definition 2.3 and Theorem 2.2 discussed in Section 2.4.2, where it discusses the satisfaction of the Restricted Isometry Property in order for the l_1- norm solution to coincide with the l_0 - norm solution. In [47], the authors proved that if

$$\omega\alpha_{\lceil 2k\omega \rceil}^{\min} + \| A \|_2 \leq \omega \tag{6.2}$$

is satisfied for any $\omega > 1$, where $\|A\|_2$ represents the Euclidean norm of A and $\lceil 2k\omega \rceil$

represents the nearest integer greater than or equal to $2k\omega$, then SL0 recovers the actual solution x^* for the l_0 -norm problem, provided that $\|x^*\|_0 = k$. As mentioned in Definition 2.3, $\alpha_k^{\min}$ represents the asymmetric k -restricted constant, which is the smallest non-negative number that satisfies the expression in Definition 2.3.

Furthermore, [47] tries to investigate the conditions with respect to $y = Ax$ where $F_\sigma(x)$ is concave near the global maximum. Therefore, by starting the steepest ascent from any point in this region will guarantee convergence to the global maximum. Initially, the parameters $\gamma_A(n_0)$ and n_0 are introduced which depend on the design matrix A and the sparsity level of x , respectively. Thereafter, a relationship is found between $\gamma_A(n_0)$ and α_k which facilitates finding the conditions to guarantee finding the sequence of σ that forces SL0 to converge to the global maximum.

6.1.1 Relationship between $\gamma_A(n_0)$ and α_k

We will define the term $\gamma_{A(n_0)}$ as described in [47] by considering a matrix $A \in \mathbb{R}^{m\times n}$. Let us first define $\pi_i(x)$ as the i^{th} element of x and idx as an ascending set of indices from a subset of x . In other words, $idx = \{i_1 < i_2 < \ldots < i_r\} \subseteq \{1,2,\ldots,n\}$ and its complement $idx^c = \{1,2,\ldots,n\} - idx$.

Definition 6.1: For a given matrix A ,

$$\begin{aligned}\gamma_A(n_0) &= \max_{|idx|\le n_0} \max_{Ax=0} \frac{\left\|\pi_{idx}(x)\right\|_2^2}{\left\|\pi_{idx^c}(x)\right\|_2^2} \\ &= \max_{|idx|\le n_0} \max_{Ax=0} \frac{\|x\|_2^2 - \left\|\pi_{idx^c}(x)\right\|_2^2}{\left\|\pi_{idx^c}(x)\right\|_2^2} \\ &= \max_{|idx|\le n_0} \max_{Ax=0} \frac{\|x\|_2^2}{\left\|\pi_{idx^c}(x)\right\|_2^2} - 1\end{aligned} \tag{6.3}$$

where $|idx|$ is the cardinality of idx .

If we let $null(A) = \{x \in \mathbb{R}^n \mid Ax = 0\}$, then any x that belongs to $null(A)$ would satisfy the following relationship:

$$\begin{aligned} Ax &= 0 \\ A_{idx}\pi_{idx} + A_{idx^c}\pi_{idx^c} &= 0 \\ \left\|A_{idx}\pi_{idx}\right\|_2 &= \left\|A_{idx^c}\pi_{idx^c}\right\|_2 \end{aligned} \tag{6.4}$$

where A_{idx} and A_{idx^c} are sub-matrices of A with columns corresponding to idx and idx^c respectively. Let us also refer $\sigma_{\min}(A_{idx})$ and $\sigma_{\max}(A_{idx^c})$ to be the smallest and the largest singular values of the sub-matrices A_{idx} and A_{idx^c} respectively. Then from (6.3) [47] derives the following two relationships:

$$\begin{aligned} \left\|A_{idx}\pi_{idx}\right\|_2 &\ge \sigma_{\min}(A_{idx})\left\|\pi_{idx}\right\|_2 \\ \left\|A_{idx^c}\pi_{idx^c}\right\|_2 &\le \sigma_{\max}(A_{idx^c})\left\|\pi_{idx^c}\right\|_2 \end{aligned}$$

Then by using Definition 6.1, (6.3) and (6.4), [47] derives the following:

(6.5)

$$\gamma_A(n_0) \leq \max_{|\mathrm{idx}|\leq n_0} \frac{{\sigma^2}_{\max}(A_{idx^c})}{{\sigma^2}_{\min}(A_{idx})}$$

Using the above arguments [47] also proves the following:

$$\gamma_A(n_0)+1 \leq \max_{|\mathrm{idx}|\leq n_0} \frac{{\sigma^2}_{\max}(A)}{{\sigma^2}_{\min}(A_{idx})} = \frac{\|A\|_2^2}{\min_{|\mathrm{idx}|\leq n_0} {\sigma^2}_{\min}(A_{idx})} \tag{6.6}$$

Finally, assuming that matrix A satisfies the Unique Representation Property (URP), using (6.2) and (6.6), [47] deduces the following relationship between $\gamma_A(n_0)$ and α_{n_0}:

$$\gamma_A(n_0)+1 \leq \frac{\|A\|_2^2}{1-{\alpha_{n_0}}^{\min}} \tag{6.7}$$

The URP property assures that any $m \times m$ sub-matrix of A is invertible. In other words, if the URP property is satisfied for A, given that $|idx| \leq m$, A_{idx} will have linearly independent columns. Therefore, as long as $|idx| \leq m$, ${\sigma^2}_{\min}(A_{idx}) > 0$ and ${\sigma^2}_{\max}(A_{idx}) < \infty$. Hence, from (6.5) we can say that $\gamma_A(n_0)$ is finite and is an increasing function of n_0.

Mohimani *et al*. in [47], then derive the following two main theorems which discuss the global convergence criterion for a general case. Theorem 6.1, which is first introduced in [46] states that if σ is chosen such that $F_\sigma(x_\sigma)$ is greater than or equal to $n-m+k$, then using the Graduated Non-convexity method the sequence of these points will converge to the sparsest solution as $\sigma \to 0$.

Theorem 6.1: Consider a family of uni-variate functions $f_\sigma : 0 \le f_\sigma(x) \le 1; \forall \sigma \in \mathbb{R}^+, x \in \mathbb{R}$, which follows the properties described in Section 3.1.1. Let $F_\sigma(x) = \sum_{i=1}^{n} f_\sigma(x_i)$, A satisfies the URP property, actual sparsest solution $x^* \in S_y (S_y = \{x \in \mathbb{R}^n \mid y = Ax\})$ satisfies $\|x^*\|_0 = k \le m/2$ and $F_\sigma(x_\sigma) \ge n - m + k$, then $\lim_{\sigma \to 0} x_\sigma = x^*$.

The above theorem highlights the importance of the condition $m - k \ge n - F_\sigma(x_\sigma)$, and since $k \le m/2$, as long as $m/2 \ge n - F_\sigma(x_\sigma)$ is satisfied for all values of σ, global maximum can be attained. Before moving on to Theorem 6.2, which discusses the selection procedure for the sequence of σ, we need to include the following Lemma from [47], which computes the bounds between two points x_1 and x_2 in the sense of Euclidean distance.

Lemma 6.1: Given that $F_{\sigma,\gamma_A(n_0)}(x_i) \ge n - \frac{n_0}{2 + 2\gamma_A(n_0)}; i = 1,2$, for two points x_1 and x_2 of S_y, the Euclidean distance between x_1 and x_2 is $\|x_1 - x_2\|_2 \le 2\sigma\sqrt{m(\gamma_A(n_0) + 1)}$.

It is evident that when $F_{\sigma,\gamma_A(n_0)}(x_i)$ surpasses a certain threshold for the given two points, the distance of the two points are bounded by $O(m^{1/2}\gamma^{1/2}\sigma)$. This relationship is

used as the basis to construct the theory and proofs by the authors in [47] for Theorem 6.2, which shows how to find a sequence of decreasing σ's such that

$F_{\sigma,\gamma_A(n_0)}(x_i) \geq n - \frac{n_0}{2+2\gamma_A(n_0)}$ is continually satisfied.

Theorem 6.2: Let us assume that A satisfies the URP and f_σ follows the properties described in Section 3.1.1. Also, let us assume that $k = \|x\|_0 \leq \frac{n_0}{2+2\gamma_A(n_0)}$, and using the minimum norm solution $x^{(0)} = (A^T A)^{-1} A^T y$ to be the initial solution for x. Let us also denote $\sigma_1 = \frac{\|x^{(0)}\|_2}{\sqrt{k(1+\gamma_A(n_0))}}$ and $c = \frac{2n}{2n + n_0/(2+2\gamma_A(n_0)) - k} < 1$. If the sequence of σ is chosen such that $\sigma_{j+1} = c\sigma_j$, and the optimization is carried out using the steepest ascent method starting from $x^{(0)}$, then at each step: $F_{\sigma_j}(x_j) \geq n - k$ and $\lim_{j\to\infty} x_j = x^*$.

Theorem 6.2 is for the noiseless case where $y = Ax$ is satisfied. For the noisy case where $S_\varepsilon = \{x \mid \|Ax - y\| < \varepsilon\}$, and ε is an arbitrary small positive number, the selection criteria for the sequence of σ's which guarantee global optimality is included in Theorem 6.3 as follows:

Theorem 6.3: Assume that A and f_σ satisfy the conditions in Theorem 6.2. Also, let $x^* \in S_\varepsilon$ to be a sparse solution and assume the condition $k < \frac{n_0}{2 + 2\gamma_A(n_0)}$ is met. Let us choose any k' which satisfies $k < k' < \frac{n_0}{2 + 2\gamma_A(n_0)}$, $\sigma_1 = \frac{\left\| x^{(0)} \right\|_2}{\sqrt{k'(1 + \gamma_A(n_0))}}$, the scale factor $c = \frac{2n}{2n + \frac{n_0}{(2 + 2\gamma_A(n_0))} - k'} < 1$, and set $\sigma_j = \sigma_1 c^{j-1}, 1 \le j \le J$, where J is the index of the smallest term of the σ satisfying $\sigma_J \ge \frac{2\sqrt{n} \,\|\, A \,\|_2 \, \varepsilon}{(1 + \gamma_A(n_0))(k' - k)} > \sigma_{J+1} = c\sigma_J$.

Then, following the steepest ascent direction and terminating at step J, we would obtain a solution that would be $C\varepsilon$ Euclidean distance away from the sparsest solution, where $C = \left(\frac{4n}{c(1 + \gamma_A(n_0))(k' - k)} + 1 \right) \|A\|_2$.

Observing the behavior of k' from Theorem 6.2 and Theorem 6.3, it is evident that choosing a suitable value for k' is of high importance. If $k' \to \frac{n_0}{2 + 2\gamma_A(n_0)}$ then $c \to 1$, and since $\sigma_{j+1} = c\sigma_j$ (from Theorem 6.2) this will result in a large number of iterations before σ_j would converge, and therefore a high computational cost. At the same time, if $k' \to k$ then $C \to \infty$, which makes the error bound to go to infinity. The authors in [47] provided a method to find a suitable k' in their final algorithm.

Even though the above convergence analysis is comprehensive in its making, finding $\gamma(n_0)$ which depends on the matrix A becomes difficult as the dimensions of the A increases. As a solution to this problem, [47] introduced almost sure upper-bounds on $\gamma(n_0)$ for large random Gaussian matrices. We will use the same concept for the convergence analysis for SGSL0, which will be discussed in the next section.

6.2 Convergence Analysis for Sparse Grouped Smoothed l_0 (SGSL0) method

As we mentioned in Section 4.3.3 in Chapter 4, the overall cost function to be minimized in (4.18) will be decomposed into sub-problems for each group l, and then each sub-problem will be minimized sequentially while keeping all the other groups fixed. We can re-write the minimization for each group l as:

$$L_{SGSL0}(x^l) = \min_{x^l} \left[\frac{1}{2} \left\| y^{-l} - A^l x^l \right\|_2^2 + \lambda_1 \left\| x^l \right\|_2 + \lambda_2 [n_l - \sum_{i=1}^{n_l} e^{\frac{-(x_i^l)^2}{2\sigma^2}}] \right] \tag{6.8}$$

It is important to note that, in Algorithm 2, finding the set of σ's which guarantees global convergence applies to the case where $x^l \neq \vec{0}$. The reason for this is that the case for $x^l = \vec{0}$ is already dealt with prior to this aforementioned step.

In order to emulate the convergence analysis described above for the SL0 case to the SGSL0 case, we combine the completely convex components of (6.8) together to form a single quadratic component. In other words, we combine the convex and differentiable fidelity component $\left\| y^{-l} - A^l x^l \right\|_2^2$ together with a quadratic approximation of $\lambda_1 \left\| x^l \right\|_2$ to form a single quadratic component. It is important to note here that since we are dealing with a quadratic approximation, which is inherently convex, we are not violating the global convergence criterion discussed in [47] (two convex functions added together will result in a convex function).

Let us refer the current iterate for x^l as x_0^l. Let us also make the important assumption here that x_0^l will lie within close proximity to the global optimality point $x^l *$, thereby not violating the condition mentioned in Theorem 6.1.

Now we can write the quadratic approximation for $\lambda_1 \left\| x^l \right\|_2$ as follows:

$$\lambda_1 \left\| x^l \right\|_2 \approx \lambda_1 \left[\left\| x_0^l \right\|_2 + \frac{\nabla^T (\left\| x_0^l \right\|_2)}{\nabla x_0^l} \Delta x^l + \frac{L}{2} \left\| \Delta x^l \right\|_2^2 \right] = \left\| g - B x^l \right\|_2^2 \tag{6.9}$$

where $\Delta x^l = x^l - x_0^l$, $x^l \in \mathbb{R}^{n_l}, g \in \mathbb{R}^{n_l}, B \in \mathbb{R}^{n_l \times n_l}$ and L is found using a line-search method as described in Section 4.3.3.

We can expand (6.9) and find g and B as follows:

$$\left(\left\|x_0^l\right\|_2-\frac{(x_0^l)^T x_0^l}{\left\|x_0^l\right\|_2}+\frac{L}{2}(x_0^l)^T x_0^l\right)+\left(\frac{(x_0^l)^T}{\left\|x_0^l\right\|_2}-L(x_0^l)^T\right)x^l+\frac{L}{2}(x^l)^T x^l = \frac{g^T g-2g^T Bx^l+(x^l)^T B^T Bx^l}{\lambda_1} \tag{6.10}$$

If we let $\eta=\left(\frac{(x_0^l)^T}{\left\|x_0^l\right\|_2}-L(x_0^l)^T\right)\in\mathbb{R}^{1\times n_l}$ then $-\frac{2}{\lambda_1}g^T B=\eta$ and $B=\sqrt{\frac{\lambda_1 L}{2}}I_{n_l}$. Therefore,

$g=-\sqrt{\frac{\lambda_1}{2L}}\eta^T$.

Now we can combine $\left\|y^{-l}-A^l x^l\right\|_2^2$ and $\left\|g-Bx^l\right\|_2^2$ to a single quadratic form as follows:

$$\left\|\begin{bmatrix}y^{-l}\\ g\end{bmatrix}-\begin{bmatrix}A^l\\ B\end{bmatrix}x^l\right\|_2^2=\left\|y^{-l}-A^l x^l\right\|_2^2+\left\|g-Bx^l\right\|_2^2 \tag{6.11}$$

If we let $Y=\begin{bmatrix}y^{-l}\\ g\end{bmatrix}\in\mathbb{R}^{(m+n_l)\times 1}$ and $M=\begin{bmatrix}A^l\\ B\end{bmatrix}\in\mathbb{R}^{(m+n_l)\times n_l}$ then we can re-write the SGSL0 cost function for a given group l as:

$$\min_{x^l}\left\|Y-Mx^l\right\|_2^2+\lambda_2\left\|x^l\right\|_0 \tag{6.12}$$

which is similar to (6.1).

It is also important to note that now the cost minimization model is an over determined system since M is a tall matrix. Therefore, since the above system is inconsistent, we use Theorem 6.3 to find the sequence of σ's which will guarantee global convergence.

Additionally, as described above, finding $\gamma(n_0)$ for a given matrix would be computationally infeasible if the dimensions of the matrix is relatively large. Nevertheless, $\gamma(n_0)$ can be derived from the ARIC's which can be found using the exhaustive process of Johnson Lindenstrauss (JL) Lemma described in [89].

As an alternative to this tedious computational process, [47] introduced upper bounds for the term $\gamma(n_0)$ for a random matrix $A \in \mathbb{R}^{m \times n}$ where $m << n$, i.e., a flat matrix. Following a similar criteria, we derive the upper bounds of $\gamma(n_0)$ for the matrix M which is a tall matrix.

In [47] it is proven that for a given random matrix $G \in$ ° where $\tilde{n} > \tilde{m}$, and $\sigma_{\max}(.)$, $\sigma_{\min}(.)$ denote the largest and smallest singular values of a given matrix:

$$P\left\{\sigma_{\max}(\sqrt{\tilde{n}/\tilde{m}}\,G) > 1+\sqrt{\tilde{n}/\tilde{m}}+r\right\} \le e^{(-\frac{\tilde{m}r^2}{2})}$$
$$P\left\{\sigma_{\min}(\sqrt{\tilde{n}/\tilde{m}}\,G) < 1-\sqrt{\tilde{n}/\tilde{m}}+r\right\} \le e^{(-\frac{\tilde{m}r^2}{2})} \quad (6.13)$$

Using (6.13), we derive the upper bounds of $\gamma(n_0)$ for the problem in (6.12) as follows:

Let idx be a subset of $\{1,...,n_l\}$, $|idx| = n_{0,l}$ and $M_{idx} \in \mathbb{R}^{(m+n_l)\times n_{0,l}}$ be a sub-matrix of M with columns corresponding to idx. Using (6.13) and [47] we can write the following probabilities:

$$P\left\{\sigma_{\max}(\sqrt{(m+n_l)/n_l}\,M) > 1+\sqrt{(m+n_l)/n_l}+\varepsilon\right\} \le e^{(-\frac{n_l\varepsilon^2}{2})}$$
$$P\left\{\sigma_{\min}(\sqrt{(m+n_l)/n_{0,l}}\,M_{idx}) < 1-\sqrt{(m+n_l)/n_{0,l}}-r\right\} \le e^{(-\frac{n_{0,l}r^2}{2})} \quad (6.14)$$

For any subset idx where $|idx| = n_{0,l}$, there can be a total of $\binom{n_l}{n_{0,l}}$ such subsets.

Following the same justifications in [47] and using the second inequality of (6.14), we can say,

$$P\left\{\sqrt{(m+n_l)/n_{0,l}}\,\min_{|idx|=n_{0,l}}\sigma_{\min}(M_{idx}) < 1-\sqrt{(m+n_l)/n_{0,l}}-r\right\} \le \binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})} \quad (6.15)$$

Now using (6.6), (6.14) and (6.15) we can write the following relation:

$$P\{\frac{n_{0,l}}{n_l}\gamma(n_{0,l}) > \frac{(1+\sqrt{(m+n_l)/n_l}+\varepsilon)^2}{(1-\sqrt{(m+n_l)/n_{0,l}}-r)^2}\} \le \binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})} + e^{(-\frac{n_l\varepsilon^2}{2})} \quad (6.16)$$

(The derivation of (6.16) is included in the Appendix section)

Then as shown in [47], using the relation $\binom{n_l}{n_{0,l}} \le \left(\frac{n_l e}{n_{0,l}}\right)^{n_{0,l}} \le e^{(n_{0,l}\log(\frac{n_l e}{n_{0,l}}))}$ where e is the Euler's number, we can obtain the following relation:

$$P\left\{\frac{n_{0,l}}{n_l}\gamma(n_{0,l}) > \frac{(1+\sqrt{(m+n_l)/n_l}+\varepsilon)^2}{(1-\sqrt{(m+n_l)/n_{0,l}}-r)^2}\right\} \le e^{(n_{0,l}\log(\frac{n_l e}{n_{0,l}}))} e^{(-\frac{n_{0,l}r^2}{2})} + e^{(-\frac{n_l\varepsilon^2}{2})} \quad (6.17)$$

$$P\left\{\frac{n_{0,l}}{n_l}\gamma(n_{0,l}) > \frac{(1+\sqrt{(m+n_l)/n_l}+\varepsilon)^2}{(1-\sqrt{(m+n_l)/n_{0,l}}-r)^2}\right\} \le e^{(n_{0,l}(\log(\frac{n_l e}{n_{0,l}})-\frac{r^2}{2}))} + e^{(-\frac{n_l\varepsilon^2}{2})}$$

Therefore, by observing the R.H.S of (6.17) we can see that if r is chosen such that $r > \sqrt{2\log(\frac{n_l e}{n_{0,l}})}$, then when $n_l \to \infty$, R.H.S of (6.17) goes to 0. Therefore, when n_l is a large number, we can compute the upper bound for $\gamma(n_{0,l})$ as

$$\left(\frac{n_l}{n_{0,l}}\right)\frac{(1+\sqrt{(m+n_l)/n_l}+\varepsilon)^2}{(1-\sqrt{(m+n_l)/n_{0,l}}-r)^2}.$$

Having the knowledge of upper bounds for the term $\gamma(n_{0,l})$ will enable us to find the conditions which satisfy (6.2) using (6.5).

6.2.1 Simulation Results

We continue to follow exactly the same steps described in part B, Section VI in [47] for the case of unknown $\gamma(n_{0,l})$ to find the sequence of σ's that guarantee global convergence. We carry out the algorithms described in Figure 3 and then Figure 2 in [47] sequentially to find the sequence of σ's, while changing $\gamma(n_{0,l})$ to the value we had computed for the SGSL0 case. We choose $\alpha = \frac{n_l}{(m+n_l)}, \beta = \frac{n_{0,l}}{(m+n_l)}$ and found β^*, which is the maximizer of $\beta/\left(2+2\gamma(n_{0,l})\right)$ on $0 \le \beta \le \alpha$. For our simulation, we choose $m = 400, n_l = 1000$. In order to satisfy the $k < \frac{n_{0,l}}{2(1+\gamma(n_{0,l}))}$ condition (Theorem

6.3), we set the sparsity k as $k = 80$. Due to the complexity of the problem, in order to find $\beta*$ we had to resort to a numerical method using the "vpasolve" function in Matlab. We used an initial guess of $\beta = 0.3$ in this computation and achieved $\beta* = 0.55$, which is between 0 and α; $\alpha = 0.714$. Finally, we attain the σ values as $\sigma_1 = 3.4138$ and $\sigma_J = 5.61\times10^{-4}$ where $\sigma_j = \sigma_1 c^{j-1}; (1 \le j \le J), J = 2633, c = 0.9924$. In other words, the initial σ to begin the outer iteration is 3.4138 and the final value for σ is 5.61×10^{-4}, where in between there will be 2631 other σ values, which are found using $\sigma_j = \sigma_1 c^{j-1}$.

By comparing this result with the experiments performed using QSL0 and SGSL0 methods in Chapter 4 and Chapter 5, we can see that the "working" values for σ in those chapters fall within the range found here. We repeated the above computations for σ with different m and n_l values, and found that the number of σ's (value for J) to be iterated for global convergence is in the range of $10^3 : 10^4$. This is a substantial amount of σ's to iterate. As mentioned in [47], the aforementioned sequence of σ's are too overbearing on the algorithm and induces unnecessary slowness. Even though these values provide a theoretical support, they are excessively pessimistic and affect the overall algorithm adversely.

Instead of using all the σ's , we chose 20 values from J which would reasonably cover the whole span of total σ values. These σ values are as follows:

[2.93, 2.51, 2.15, 1.85, 1.59, 1.36, 1.00, 0.63, 0.54, 0.34, 0.21, 0.18, 0.16, 0.13, 0.10, 0.07, 0.05, 0.03, 0.02 and 0.01]

The results were very much similar to what we obtained in Chapter 5 (Figure 5.4 (c)) where we used the set of σ's as $\sigma = [5,1,0.7,0.5,0.3,0.1]$. Therefore, we can conclude that since the order and the magnitude of the "working" σ values from Chapter 5 are very similar to the theoretically obtained values here (even though the final result has negligible improvement for very small σ values), it justifies the use of a few selected σ's instead of iterating in its entirety.

7 REGULARIZATION PARAMETER SELECTION IN SPARSE GROUP SL0 (SGSL0) USING MODEL SELECTION

Finding appropriate regularization parameters is important since they largely affect the performance of the predicted model. The regularization parameters determine the level of impact each term has on the overall cost function and it differs from one solution to another. For this purpose, we expect to follow a model selection criterion based on the Generalized Information Criteria (GIC) [81], which was formulated by Shimamura et al. in [76]. This criterion was used to select the best regularization parameter from a set of candidate values, for the Group Lasso framework.

For a given set of models with different regularization parameters each, the best model will be the one with the least Kullback-Leibler information [82]. Kullback-Leibler information measures the divergence between a probability density function of an unknown distribution and its predictive density function for a future observation. Using this formulation as our basis, we plan to extend Shimamura *et al*.'s work[76] to find the Information Criterion to select the best set of regularization parameters from a set of candidate values for the case of Sparse Group SL0 (SGSL0) method. This criterion can be especially useful for selecting the free parameters when we have limited *a-priori* knowledge about the original signal to be reconstructed.

In this section we plan to briefly introduce the formulation of the theory introduced by Shimamura et al. in [76]. We have omitted the intermediate steps of most of the theory, which can be referred to [76, 81].

7.1 Generalized Information criteria in model selection

Suppose Y_m is a random sample of size m from an unknown distribution $G(y)$ having a probability density function $g(y)$. The parametric family of distributions used for predictions are represented by $\{f(y|\theta), \theta \in \Theta\}$, which may or may not contain g. Here θ is an unknown vector of parameters of length n and the predictive density $f(z|\hat{\theta})$ for a future observation z can be constructed by using an estimation vector $\hat{\theta}$. Suppose that $\hat{G}$ represents an empirical distribution substituting the unknown distribution G.

Using the above notations, the Information Criteria mentioned in [76] can be written as follows: (refer [76] and [81] for intermediate steps)

$$IC = -2\sum_{i=1}^{m} \log f\left(y_i|\hat{\theta}\right) + 2\hat{b}(G) \tag{7.1}$$

where $\hat{b}(G)$ represents the approximation for the bias term given by

$$b(G) = E_{G(y)}\left[\int \log f(z|\hat{\theta}) d\hat{G}(z) - \int \log f(z|\hat{\theta}) dG(z)\right] \tag{7.2}$$

with the expectation taken over the joint distribution of $y : \prod_{(i=1)}^{m} dG(y_i)$.

For a general statistical functional estimator $\hat{\theta} = T(\hat{G})$, where $T(.)$ is a functional vector on the space of distribution functions with dimension n, $\hat{b}(G)$ is derived as [81]:

$$\hat{b}(G) = \frac{1}{m} tr \left\{ \int T^{(1)}(z;G) \frac{\partial \log f(z|\theta)}{\partial \theta^T} |_{\theta=T(G)} dG(z) \right\} \tag{7.3}$$

where $T^{(1)}(z;G)$ is defined with respect to the influence functions $T_i^{(1)}(z;G)$ as follows:

$$T^{(1)}(z;G) = \left(T_1^{(1)}(z;G), \ldots, T_n^{(1)}(z;G) \right)^T \tag{7.4}$$

$$T_i^{(1)}(z;G) = \lim_{\varepsilon \to 0} \frac{T_i\left[(1-\varepsilon)G + \varepsilon \delta_z \right] - T_i(G)}{\varepsilon} \tag{7.5}$$

$$\frac{\partial}{\partial \theta^T} = \left(\frac{\partial}{\partial \theta_1}, \ldots, \frac{\partial}{\partial \theta_n} \right) \tag{7.6}$$

Here T_i is the i^{th} component of T and δ_z is a point mass at z .Therefore, by replacing $\hat{b}(G)$ with $\hat{b}(\hat{G})$ we can obtain the information criterion in (7.1).

Let us overview the works of [76] by considering the objective function for the Group Lasso model as follows:

$$\frac{1}{2} \left\| y - \sum_{j=1}^{J} A^j x^j \right\|_2^2 + \lambda \sum_{j=1}^{J} \left\| x^j \right\|_2 \tag{7.7}$$

where $y \in \mathbb{R}^m$, $x \in \mathbb{R}^n = \left[x^1,\ldots,x^J\right]^T$. Let the length of a given group (all groups are non-overlapping) x^j be n_j, and therefore, $\sum_{j=1}^{J} n_j = n$. Let $A \in \mathbb{R}^{m\times n}$ be divided into sub-matrices corresponding to the groups of x as follows: $A = \left[A^1 A^2 \ldots A^J\right]$, where A^j is an m by n_j matrix.

Then in [76], $\hat{b}(G)$ is computed using the following Gaussian model:

$$f\left(y_i|\hat{\theta}\right) = \frac{1}{\sqrt{2\pi\hat{\kappa}^2}} \exp\left\{-\frac{\left(y_i - a_i^T\hat{x}\right)^2}{2\hat{\kappa}^2}\right\} \tag{7.8}$$

where κ is the standard deviation, a_i^T is the i^{th} row of A and $\hat{\theta} = (\hat{x}^T, \hat{\kappa}^2)^T$. Once the information criterion is computed for each model with a specific λ, the model with the least information is then selected to have the best regularization parameter.

7.2 Regularization parameter selection for SGSL0 method

We will use similar arguments to the SGSL0 model where the minimization of the objective function is defined as:

$$\min_x L_{SGSL0}(x) = \min_x \left[\frac{1}{2}\left\|y - \sum_{j=1}^{J} A^j x^j\right\|_2^2 + \lambda_1 \sum_{j=1}^{J} \left\|x^j\right\|_2 + \lambda_2 \sum_{j=1}^{J} [n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}}]\right] \tag{7.9}$$

where $\lambda_1, \lambda_2 > 0$ are the regularization parameters. We can consider l^* such candidate models with l^* different pairs of (λ_1, λ_2), and compute the IC for each model accordingly. The hypothesis would be that the lowest IC would yield the best reconstruction.

Now using (7.8) we can write the "penalized log-likelihood function" as:

$$l_{\lambda_1,\lambda_2}(y \mid \theta) = \sum_{i=1}^{m} \log\left(f\left(y_i \mid \theta\right)\right) - \frac{\lambda_1}{\kappa^2} \sum_{j=1}^{J} \left\| x^j \right\|_2 - \frac{\lambda_2}{\kappa^2} \sum_{j=1}^{J} \left[n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}} \right] \tag{7.10}$$

where $\sigma \to 0$.

We can also write the log likelihood function itself as:

$$\kappa^2 \sum_{i=1}^{m} \log\left(f\left(y_i \mid \theta\right)\right) = -\frac{1}{2} \sum_{i=1}^{m} \left\| y - Ax \right\|_2^2 - \frac{m\kappa^2}{2} \log(2\pi\kappa^2) \tag{7.11}$$

Therefore, if we multiply (7.10) by κ^2 and then substitute (7.11) in (7.10), we get

$$\begin{aligned} \kappa^2 l_{\lambda_1,\lambda_2}(y \mid \theta) = -\frac{1}{2} \left\| y - Ax \right\|_2^2 - \frac{m\kappa^2}{2} \log(2\pi\kappa^2) - \lambda_1 \sum_{j=1}^{J} \left\| x^j \right\|_2 \\ - \lambda_2 \sum_{j=1}^{J} [n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}}] \end{aligned} \tag{7.12}$$

It is obvious that the maximization of (7.12) w.r.t x is the same as the minimization problem in (7.9).

Once we obtain the estimated solution $\hat{x}$ using the SGSL0 algorithm, next task is to obtain the estimated standard deviation $\hat{\kappa}$. This can be obtained by finding the solution to the following [76]:

$$\frac{\partial}{\partial \kappa^2}\left\{\begin{array}{l} -\frac{1}{2\kappa^2}\left\|y-Ax\right\|_2^2-\frac{m}{2}\log(2\pi\kappa^2)-\frac{\lambda_1}{\kappa^2}\sum_{j=1}^{J}\left\|x^j\right\|_2- \\ \qquad\qquad \frac{\lambda_2}{\kappa^2}\sum_{j=1}^{J}[n_j-\sum_{i=1}^{n_j}e^{\frac{-(x_i^j)^2}{2\sigma^2}}] \end{array}\right\}=0 \tag{7.13}$$

Once we have $\hat{\theta}$ found as described above, our next step is to find the influence function $T^{(1)}\left(y;G\right)$ for $\hat{\theta}$. Unfortunately, (7.9) is not differentiable in terms of θ when some group-wise components of θ are exactly zero in the solution. To overcome this difficulty, we make the same assumption that is being made by the authors in [76]. Let us first index the groups of x as $\{1,2,...,J\}$ and denote the subset of the non-zero groups as $\xi_k=\left\{j\in\left\{1,2,...,J:\hat{\theta}^j\neq\vec{0}\right\}\right\}$ for the k^{th} (λ_1,λ_2) pair; $k=1,2,...,r$. We assume that ξ_k is locally convergent w.r.t y. In other words, it is assumed that the zero group-wise components stay the same when a small perturbation of ε is imposed on the observation vector y. As described in [76], this enables the penalized log-likelihood function to be twice differentiable w.r.t θ_{ξ_k} where $\theta_{\xi_k}=(x_{\xi_k}^T,\kappa_k^2)^T, x_{\xi_k}\in\mathbb{R}^{n_{\xi_k}}$.

Also, let us denote A_{ξ_k} to be the sub matrix of A with columns corresponding to ξ_k. Following the definitions in [76], we will define the functional vector $T_{\xi_k}(G)\in\mathbb{R}^{n_{\xi_k}\times 1}$ as:

$$\int\psi_{\xi_k}(y,\theta_{\xi_k})\left|_{\theta_{\xi_k}=T_{\xi_k}(G)}dG=0\right. \tag{7.14}$$

where

$$\psi_{\xi_k}(y_i,\theta_{\xi_k})=\frac{\partial}{\partial\theta_{\xi_k}}\left\{\begin{array}{r}-\frac{1}{2\kappa^2}(y_i-a_i^Tx)^2-\frac{1}{2}\log(2\pi\kappa^2)-\frac{\lambda_1^k}{\kappa^2}\sum_{j=1}^{J}\left\|x^j\right\|_2 \\ -\frac{\lambda_2^k}{\kappa^2}\sum_{j=1}^{J}[n_j-\sum_{i=1}^{n_j}e^{\frac{-(x_i^j)^2}{2\sigma^2}}]\end{array}\right\} \tag{7.15}$$

with G being the true distribution of y and a_i^T being the i^{th} row of A.

Let us now replace G by the empirical distribution $\hat{G}$ based on the observations y, and then using (7.14) we can have

$$\frac{1}{m}\sum_{i=1}^{m}\psi_{\xi_k}(y_i,\theta_{\xi_k})\,|_{\hat{\theta}_{\xi_k}=T_{\xi_k}(\hat{G})}=0 \tag{7.16}$$

Now following the procedure in [81], we replace G in (7.14) by $(1-\varepsilon)G+\varepsilon\delta_y$ where δ_y is a point of mass at y. Then we can re-write (7.14) as,

$$\underbrace{\int\psi_{\xi_k}(y,T_{\xi_k}((1-\varepsilon)G+\varepsilon\delta_y))\,|_{\theta_{\xi_k}=T_{\xi_k}(G_\varepsilon)}}_{E}\underbrace{d((1-\varepsilon)G+\varepsilon\delta_y))}_{F}=0 \tag{7.17}$$

Now we employ the product and chain rules to differentiate the above w.r.t ε and set $\varepsilon=0$:

$$\frac{d(EF)}{d\varepsilon}=(\frac{dE}{d\varepsilon})F+E(\frac{dF}{d\varepsilon})=\frac{dE}{d\theta}\frac{d\theta}{d\varepsilon}F+E(\frac{dF}{d\varepsilon})$$

Therefore, by substituting the above with (7.17) we get,

$$\begin{aligned}\int\frac{\partial}{\partial\theta_{\xi_k}}\psi_{\xi_k}(y,\theta_{\xi_k})\,|_{\theta_{\xi_k}=T_{\xi_k}(G)}dG(y)\frac{\partial}{\partial\varepsilon}\{T_{\xi_k}((1-\varepsilon)G+\varepsilon\delta_y)\}\,|_{\varepsilon=0}\\ +\int\psi_{\xi_k}(y,T_{\xi_k}(G))\,d\{\delta_y(y)-G(y)\}=0\end{aligned} \tag{7.18}$$

But we know that from (7.6) that the influence function can be written as

$$T_{\xi_k}^{(1)}(y,G)=\frac{\partial}{\partial\varepsilon}\{T_{\xi_k}((1-\varepsilon)G+\varepsilon\delta_y)\}\,|_{\varepsilon=0} \tag{7.19}$$

Hence using (7.18) [76][81],

$$T_{\xi_k}^{(1)}(y,G) = -\left\{\int \frac{\partial}{\partial \theta_{\xi_k}} \psi_{\xi_k}(y,\theta_{\xi_k})\,|_{\theta_{\xi_k}=T_{\xi_k}(G)} dG(y)\right\}^{-1} \psi_{\xi_k}(y,\theta_{\xi_k}) \tag{7.20}$$

Therefore, now substituting (7.20) into (7.3) we get,

$$\hat{b}(G) = \frac{1}{m} tr\left\{J(G)^{-1} I(G)\right\} \tag{7.21}$$

where,

$$J(G) = -\left\{\int \frac{\partial}{\partial \theta_{\xi_k}} \psi_{\xi_k}(y,\theta_{\xi_k})\,|_{\theta_{\xi_k}=T_{\xi_k}(G)} dG(y)\right\} \tag{7.22}$$

and

$$I(G) = \int \frac{\partial \log f\left(y \mid \theta_{\xi_k}\right)}{\partial \theta'_{\xi_k}} \psi_{\xi_k}(y, T_{\xi_k}(G)) dG(y) \tag{7.23}$$

Now using (7.1) we can re-write IC as [81],

$$IC = -2\sum_{i=1}^{m} \log f(y_i \mid \hat{\theta}) + \frac{2}{m}\sum_{i=1}^{m} tr\left\{T_i^{(1)}\left(y_i; \hat{G}\right) \frac{\partial \log f\left(y_i|\hat{\theta}\right)}{\partial \theta'}\Big|_{\theta=\hat{\theta}}\right\} \tag{7.24}$$

and therefore,

$$IC_{SGSL0} = m\log(2\pi\kappa^2) + \frac{\|y - Ax\|_2^2}{\hat{\kappa}^2} + 2tr\left\{I(\hat{G})J(\hat{G})^{-1}\right\} \tag{7.25}$$

Let us now find $J(G)$ and $I(G)$. From (7.15) we can write $\psi_{\xi_k}(y_i, \theta_{\xi_k})$ as

$$\psi_{\xi_k}(y_i,\theta_{\xi_k}) = \begin{bmatrix} \frac{1}{\hat{\kappa}^2}(a_{\xi_k})(y_i - a_{\xi_k}^T x) - \frac{\lambda_1^k \bar{d}}{\hat{\kappa}^2} - \frac{\lambda_2^k \bar{c}}{\hat{\kappa}^2} \\ \\ \frac{1}{2\kappa^2}(y_i - a_{\xi_k}^T x)^2 - \frac{1}{2\hat{\kappa}^2} + \frac{\lambda_1^k \sum_{j=1}^{J} \| x^j \|_2}{\hat{\kappa}^4} + \frac{\lambda_2^k \sum_{j=1}^{J}[n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}}]}{\hat{\kappa}^4} \end{bmatrix}_{(n_\xi+1)\times 1}$$

(7.26)

where $a_{\xi_k}^T \in \mathbb{R}^{1\times n_{\xi_k}}$ is a row of A_{ξ_k}, $\bar{d} = (\bar{d}_j)_{j\in\xi_k} \in \mathbb{R}^{n_{\xi_k}\times 1}, \bar{d}_j = \frac{x^j}{\left\|x^j\right\|_2}$ and

$\bar{c} = \frac{1}{\sigma^2} W(x_{\xi_k}) x_{\xi_k} \in \mathbb{R}^{n_{\xi_k}\times 1}$ and $W(x_{\xi_k}) = diag(e^{-x^t_{\xi_k}})$ where $t = 1 : n_{\xi_k}$. Also, the 1st and 2nd terms of $\psi_{\xi_k}(y_i, \theta_{\xi_k})$represent $\frac{\partial(.)}{\partial x_{\xi_k}}$ and $\frac{\partial(.)}{\partial \kappa^2}$ respectively.

Now let us differentiate (7.26) again w.r.t θ_{ξ_k} where the resulting matrix would have the terms: $\frac{\partial(.)}{\partial x_{\xi_k}(\partial x_{\xi_k})^T}, \frac{\partial(.)}{\partial x_{\xi_k}\partial \hat{\kappa}^2}$ as the first row and $\frac{\partial(.)}{\partial \hat{\kappa}^2(\partial x_{\xi_k})^T}, \frac{\partial(.)}{\partial \hat{\kappa}^2 \partial \hat{\kappa}^2}$ as the second row respectively,

$$\frac{\partial}{\partial \theta_{\xi_k}}(\psi_{\xi_k}(y_i, \theta_{\xi_k})) = \begin{bmatrix} \frac{-1}{\hat{\kappa}^2} a_{\xi_k}(a_{\xi_k})^T - \frac{\lambda_1^k \bar{D}}{\hat{\kappa}^2} - \frac{\lambda_2^k \bar{C}}{\hat{\kappa}^2} & \frac{-1}{\hat{\kappa}^4}(a_{\xi_k,i})(y_i - a_{\xi_k}^T x) + \frac{\lambda_1^k \bar{d}}{\hat{\kappa}^4} + \frac{\lambda_2^k \bar{c}}{\hat{\kappa}^4} \\ \frac{-1}{\hat{\kappa}^4}(y_i - a_{\xi_k}^T x)(a_{\xi_k})^T + \frac{\lambda_1^k \bar{d}^T}{\hat{\kappa}^4} + \frac{\lambda_2^k \bar{c}^T}{\hat{\kappa}^4} & \frac{-1}{\hat{\kappa}^6}(y_i - a_{\xi_k}^T x)^2 + \frac{1}{2\hat{\kappa}^4} - \frac{\lambda_1^k \sum_{j=1}^{J} \left\|x^j\right\|_2}{\hat{\kappa}^6} - \frac{\lambda_2^k \sum_{j=1}^{J}[n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}}]}{\hat{\kappa}^6} \end{bmatrix}$$

(7.27)

where $\frac{\partial}{\partial \theta_{\xi_k}}\left(\psi_{\xi_k}(y_i, \theta_{\xi_k})\right) \in \mathbb{R}^{(n_{\xi_k}+1)\times(n_{\xi_k}+1)}$,

$\bar{D} = blockdiag(\bar{D}_j)_{j\in\xi_k} \in \mathbb{R}^{n_{\xi_k}\times n_{\xi_k}}, \bar{D}_j = \frac{I_{n_j}}{\left\|x^j\right\|_2} - \frac{x^j (x^j)^T}{\left\|x^j\right\|_2^3}$ and

$\bar{C} = \frac{1}{\sigma^2} W(x_{\xi_k}) \left[I_{n_{\xi_k}} - \frac{\tilde{x}_{\xi_k}}{\sigma^2} \right] \in \mathbb{R}^{n_{\xi_k} \times n_{\xi_k}}, \tilde{x}_{\xi_k} = diag\left((x_{\xi_k}^t)^2 \right)$ where $t = 1 : n_{\xi_k}$. Note that here

n_j is the length of group j , n_{ξ_k} is the length of the set ξ_k and I represents the

Identity matrix.

Therefore, using (7.27) and following the workings of [76] we can obtain

$$J(\hat{G}) = \frac{1}{m} \begin{bmatrix} \frac{1}{\hat{\kappa}^2} A_{\xi_k}^T A_{\xi_k} + \frac{\lambda_1^k \bar{D}}{\hat{\kappa}^2} + \frac{\lambda_2^k \bar{C}}{\hat{\kappa}^2} & \frac{1}{\hat{\kappa}^4} A_{\xi_k}^T \Lambda 1_m - \frac{\lambda_1^k \bar{d}}{\hat{\kappa}^4} - \frac{\lambda_2^k \bar{c}}{\hat{\kappa}^4} \\ \frac{1}{\hat{\kappa}^4} 1_m^T \Lambda A_{\xi_k} - \frac{\lambda_1^k \bar{d}^T}{\hat{\kappa}^4} - \frac{\lambda_2^k \bar{c}^T}{\hat{\kappa}^4} & \\ \frac{1}{\hat{\kappa}^6} \left\| y - A_{\xi_k} x \right\|_2^2 - \frac{1}{2\hat{\kappa}^4} + \frac{\lambda_1^k \sum_{j=1}^{J} \left\| x^j \right\|_2}{\hat{\kappa}^6} + \frac{\lambda_2^k \sum_{j=1}^{J} [n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}}]}{\hat{\kappa}^6} & \end{bmatrix} \quad (7.28)$$

where $J(\hat{G}) \in \mathbb{R}^{(n_{\xi_k}+1)\times(n_{\xi_k}+1)}$, $\Lambda = diag(y_i - x^T a_i) \in \mathbb{R}^{m \times m} : i = 1 : m$, $1_m = (1,1,...,1)^T \in \mathbb{R}^{m \times 1}$

Likewise, using (7.11) and (7.23) we can find $I(\hat{G})$ as follows:

$$I(\hat{G}) = \frac{1}{m} \sum_{i=1}^{m} \frac{\partial \log f\left(y_i \mid \theta_{\xi_k} \right)}{\partial \theta'_{\xi_k}} \psi_{\xi_k}(y_i, T_{\xi_k}(\hat{G})) \quad (7.29)$$

from (7.11) we get,

$\log f(y_i \mid \theta) = -\frac{1}{2\kappa^2}(y_i - a_{\xi_k}^T x)^2 - \frac{1}{2}\log(2\pi\kappa^2)$ and

$$\frac{\partial \log f(y_i \mid \theta)}{\partial \theta^T} = \begin{bmatrix} \frac{(y_i - x^T a_{\xi_k,i}) a_{\xi_k,i}}{\kappa^2} & \frac{1}{2\kappa^4}(y_i - x^T a_{\xi_k,i})^2 - \frac{1}{2\kappa^2} \end{bmatrix}.$$

Therefore,

$$I(\hat{G}) = \begin{bmatrix} \frac{1}{\hat{\kappa}^2} A_{\xi_k}^T \Lambda - \frac{\lambda_1^k \bar{d} 1_m^T}{\hat{\kappa}^2} - \frac{\lambda_2^k \bar{c} 1_m^T}{\hat{\kappa}^2} \\ \frac{1}{2\hat{\kappa}^4} 1_m^T \Lambda^2 - \frac{1}{2\hat{\kappa}^2} 1_m^T + \frac{\lambda_1^k \sum_{j=1}^{J} \| x^j \|_2}{\hat{\kappa}^4} 1_m^T + \\ \frac{\lambda_2^k \sum_{j=1}^{J} [n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}}]}{\hat{\kappa}^4} 1_m^T \end{bmatrix}_{(n_{\xi_k}+1)\times m} \begin{bmatrix} \Lambda A_{\xi_k} & \frac{1}{2\hat{\kappa}^4} \Lambda^2 1_m - \frac{1}{2\hat{\kappa}^2} 1_m \end{bmatrix}_{m\times(n_{\xi_k}+1)}$$

(7.30)

7.3 Simulation Studies

In order to find the best (λ_1, λ_2) pair among a set of l' candidate set of pairs, we choose a suitable range of λ_1's and λ_2's and tabulate the corresponding IC_{SGSL0} values accordingly. The best pair should be having the lowest IC_{SGSL0} value.

For this experiment we used the SGSL0 algorithm with the sparsity level $k = 80$, the number of sensors $n = 380$, the number of sources $m = 800$, and the number of groups $J = 8$ where each group will be having an equal length of $n_j = 100 : j = 1 : J$. The non-zero groups were placed such that they belong to the set $\xi_k = \{j \in 1,...,J : x^j \neq \vec{0}\} = \{x^1, x^3, x^4, x^6\}$ where x^j represents the j^{th} group of x. In order to compare the IC_{SGSL0} values with the performance, we plot the reconstructed

signal against the original signal. The IC_{SGSL0} values for a range of (λ_1, λ_2) pairs are as follows:

λ_1	λ_2	$IC_{SGSL0} \times 10^4$
0.001	15	4.12
0.01	7	3.87
0.1	5	2.79
1	5	2.85
5	5	3.61
0.1	0.5	9.05
1	0.5	9.43
5	0.5	12.29
0.1	0.05	23.65
1	0.05	31.56
5	0.05	39.02

Table 7.1: IC_{SGSL0} values for different (λ_1, λ_2) pairs

Now for visual comparison we plot the corresponding reconstructed signals as follows:

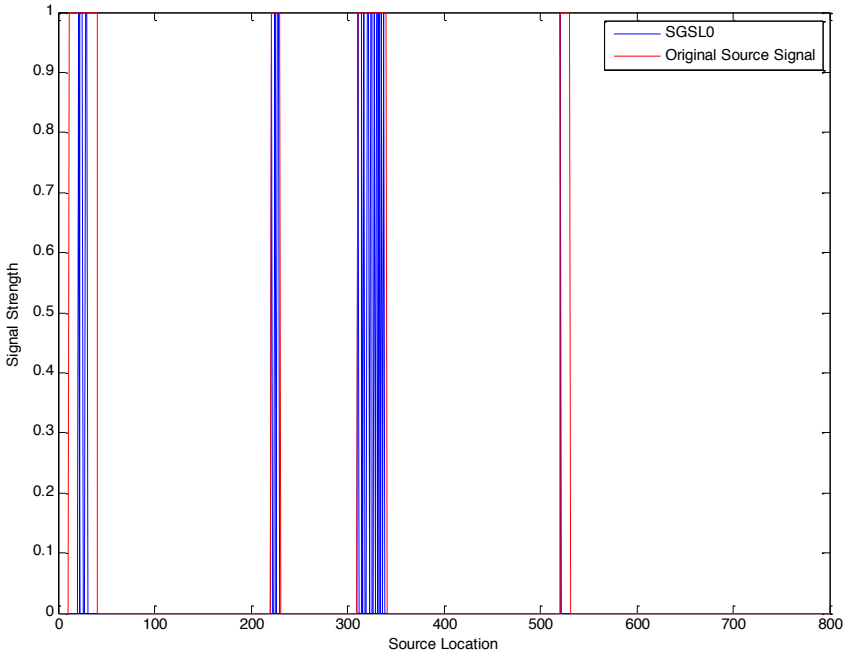


Figure7.1: Reconstructed Signal for $\lambda_1 = 0.001, \lambda_2 = 15, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 4.12$

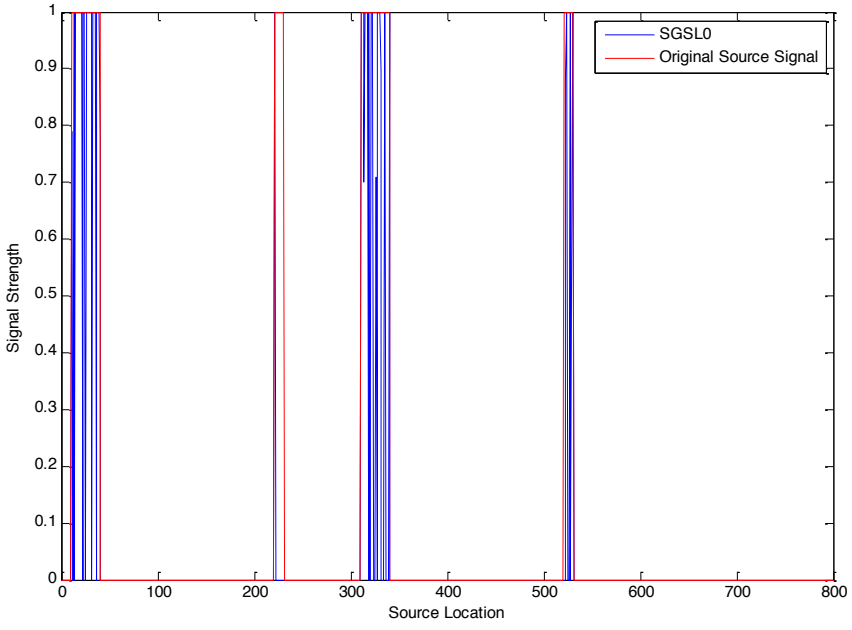


Figure7.2: Reconstructed Signal for $\lambda_1 = 0.01, \lambda_2 = 7, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 3.87$

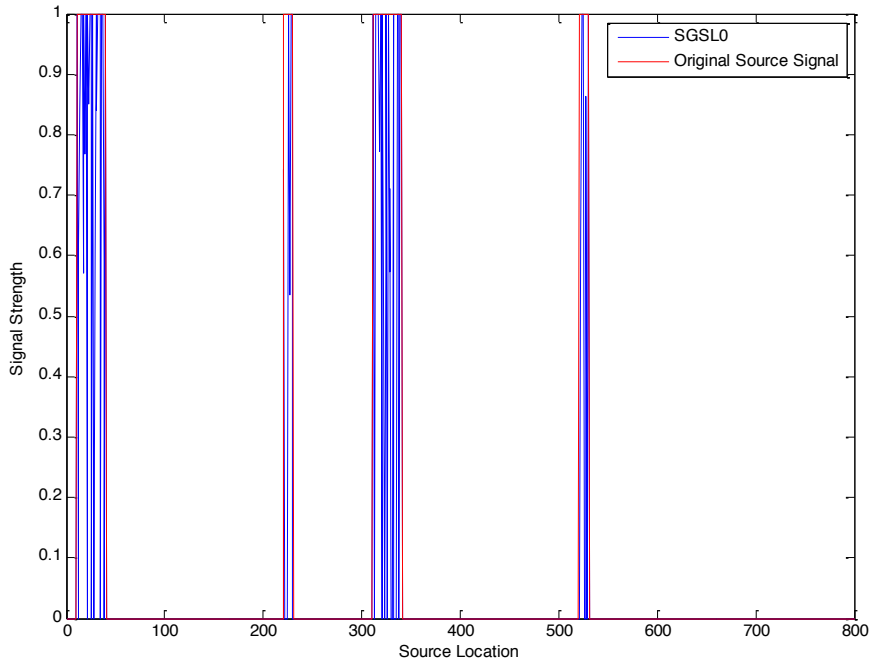


Figure7.3: Reconstructed Signal for $\lambda_1 = 0.1, \lambda_2 = 5, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 2.79$

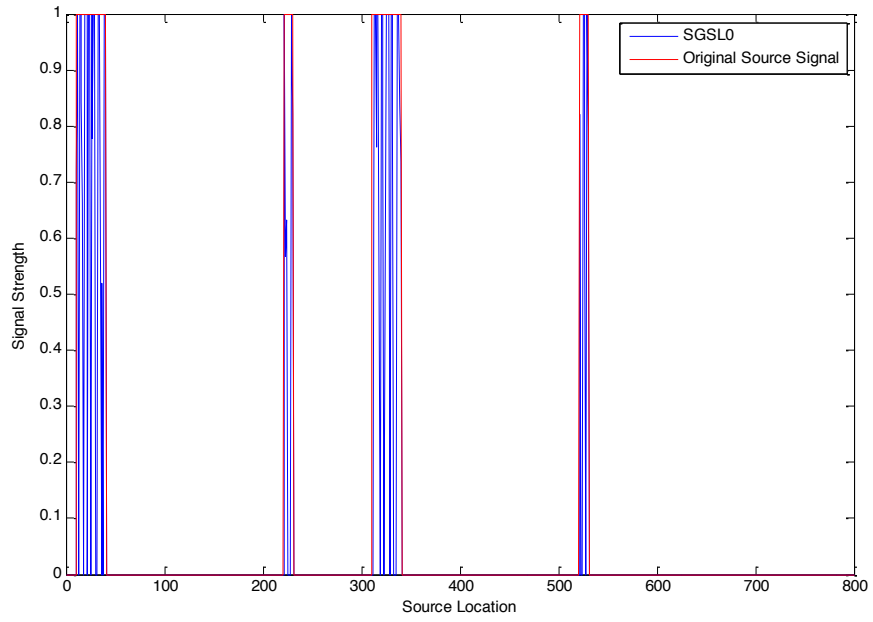


Figure7.4: Reconstructed Signal for $\lambda_1 = 1, \lambda_2 = 5, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 2.85$

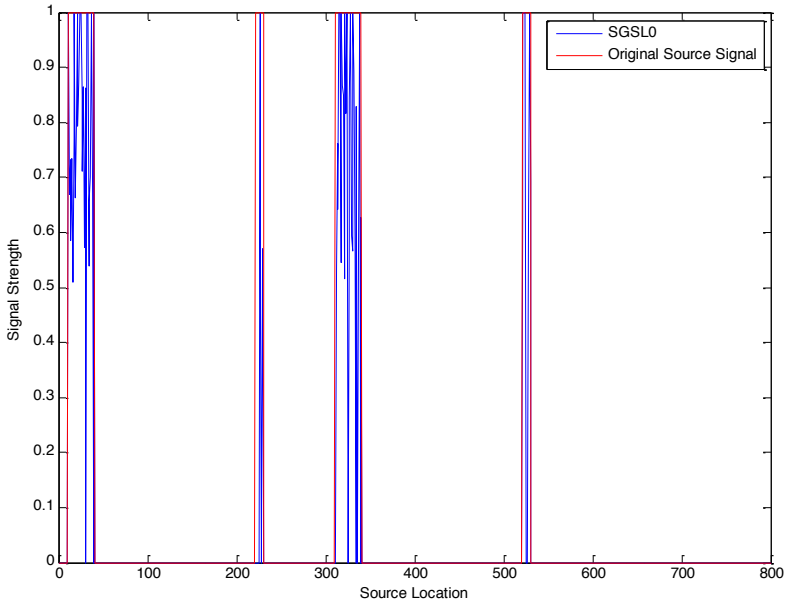


Figure7.5: Reconstructed Signal for $\lambda_1 = 5, \lambda_2 = 5, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 3.61$

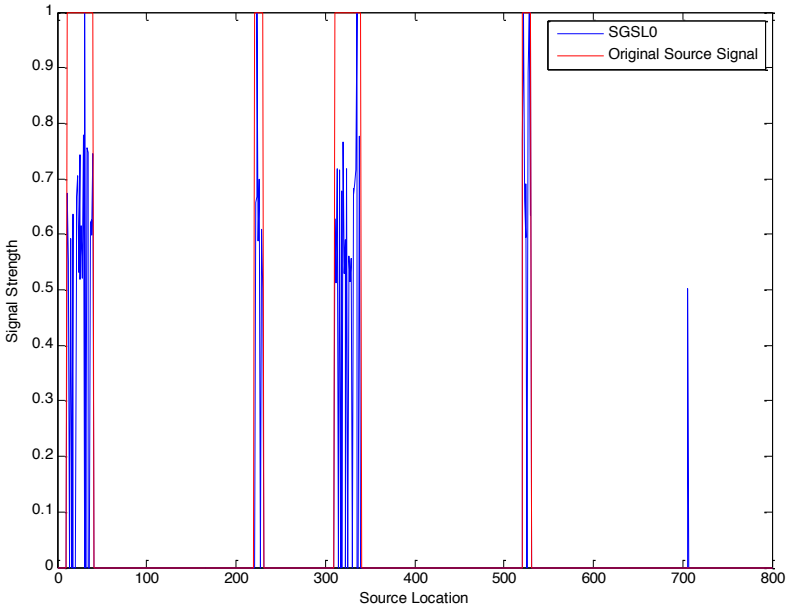


Figure7.6: Reconstructed Signal for $\lambda_1 = 0.1, \lambda_2 = 0.5, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 9.05$

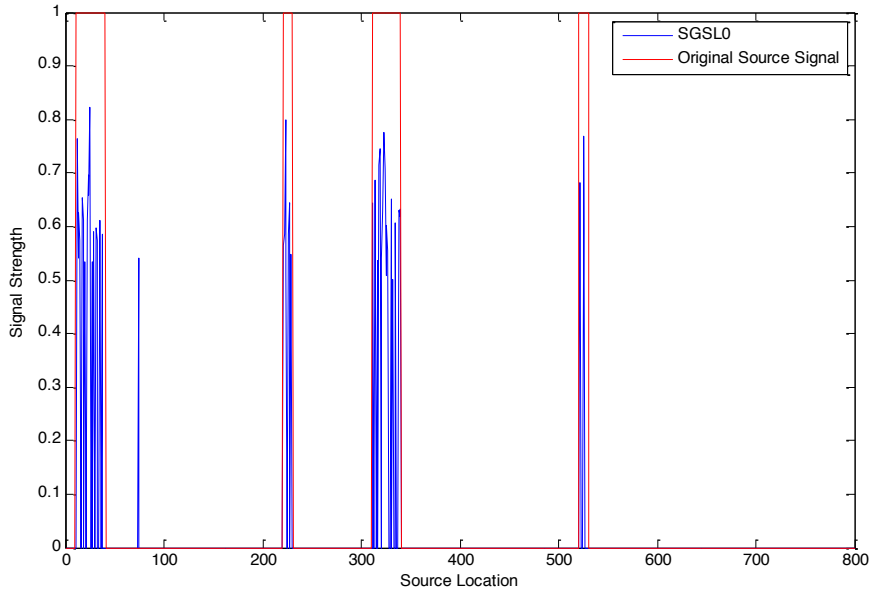


Figure7.7: Reconstructed Signal for $\lambda_1 = 1, \lambda_2 = 0.5, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 9.43$

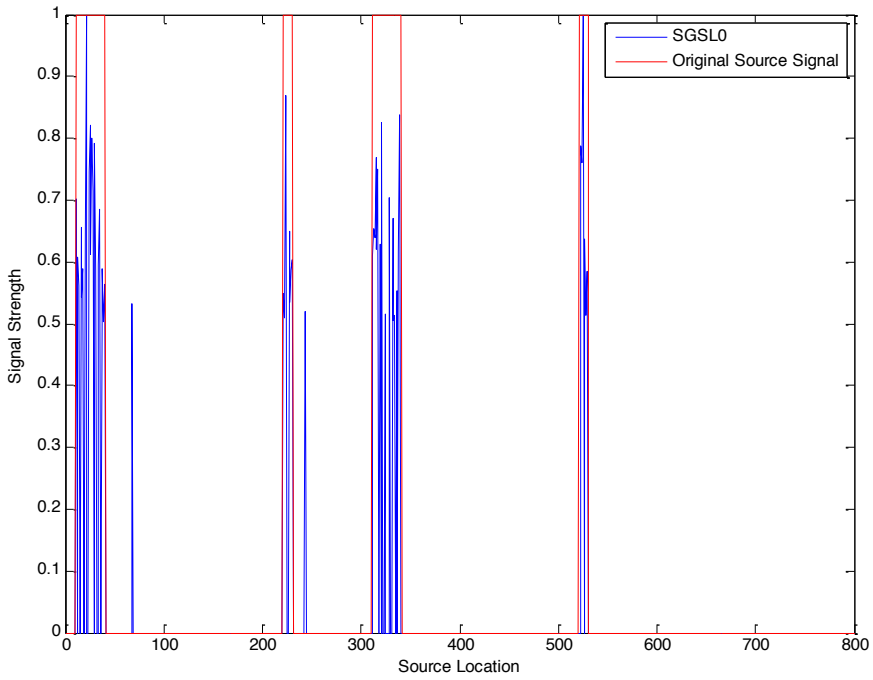


Figure7.8: Reconstructed Signal for $\lambda_1 = 5, \lambda_2 = 0.5, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 12.29$

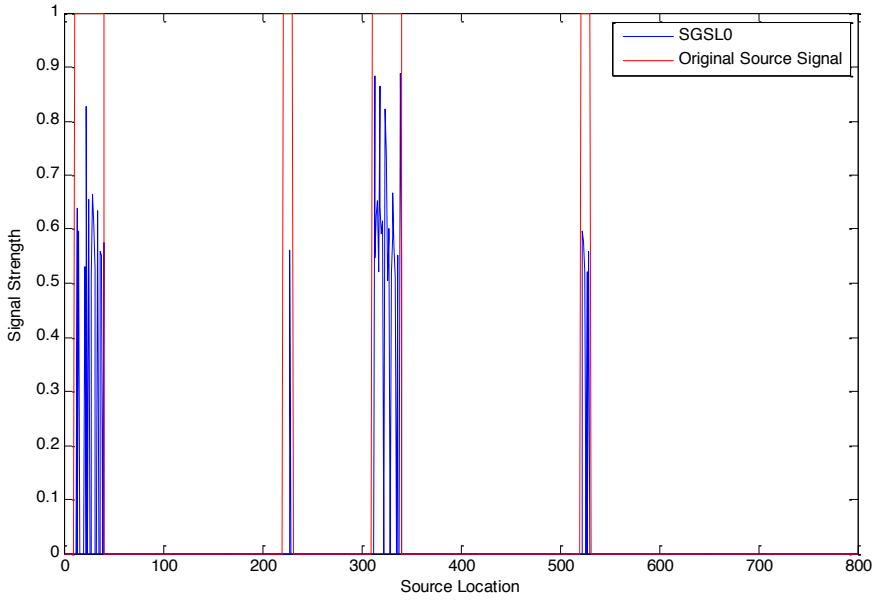


Figure7.9: Reconstructed Signal for $\lambda_1 = 0.1, \lambda_2 = 0.05, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 23.65$

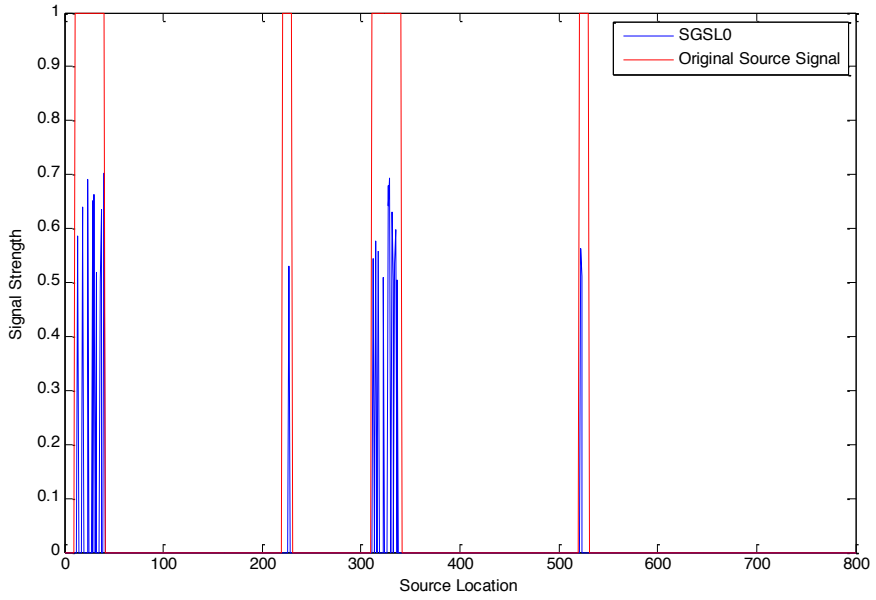


Figure7.10: Reconstructed Signal for $\lambda_1 = 1, \lambda_2 = 0.05, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 31.56$

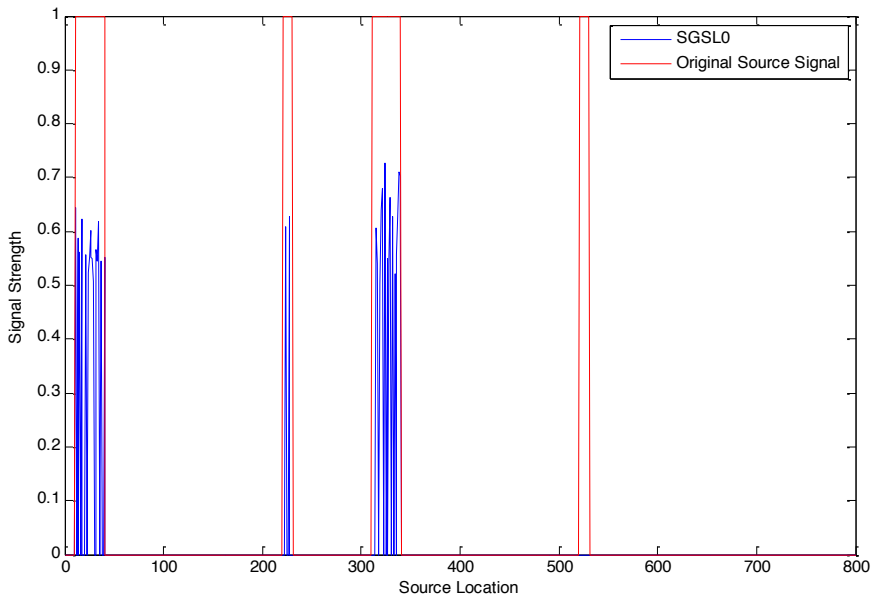


Figure7.11: Reconstructed Signal for $\lambda_1 = 5, \lambda_2 = 0.05, \mathrm{IC}_{SGSL0}\left(\mathrm{x}10^4\right) = 39.02$

Therefore, as we can see when the IC_{SGSL0} increases, the signal reconstruction becomes deteriorated. From observing the behavior of λ_1 and λ_2, we can say that the best combination of regularization parameters for this experiment would be $\lambda_1 = 0.1, \lambda_2 = 5$.

8 STOPPING CRITERION AND OPTIMALITY CONDITIONS

Usually, when the cost function needed to be optimized is smooth/ differentiable, a natural stopping criterion can be admitted based on the gradient of the cost function. For a given smooth cost function $L(x)$, for a given threshold ε, the stopping criterion could be defined as:

$$\left\|\nabla L(x^{(k)})\right\| < \varepsilon \tag{8.1}$$

The algorithm can be terminated at the k^{th} iteration which satisfies the above condition.

Unfortunately, this criterion is not valid when the cost function has a non-differentiable component. The reason is the existence of a sub-differential $\partial L(x)$ at the non-differentiable point of the function. The elements of $\partial L(x)$ are referred to as the sub-gradients of L at x. Therefore, if L is a convex function, the condition: $0 \in \partial L(x)$ satisfies the global optimality condition at point x. If the function L is differentiable at the global optimality point x, the set $\partial L(x)$ is actually the singleton $\{\nabla L(x)\}$. Hence, the condition $0 \in \partial L(x)$ reduces to the aforementioned condition in (8.1) to $\nabla L(x) = 0$. As we know, when the function L is differentiable or when there is a singleton gradient $\left(\nabla L(x)\right)$, we can use the criterion $\nabla L(x) = 0$ as the stopping criterion for an

optimization problem. But, when there is a non-differentiable cost function to be optimized, the above criterion would not work, as there will be multiple sub-gradients at the non-differentiable point to consider.

In our cost function to be minimized for the Sparse Group SL0 (SGSL0) method ((4.18)), since it is not smooth at $x^l = \vec{0}$, $\left\| x^l \right\|_2$ component imposes non-differentiability. In order to find the stopping criterion for such problems, the concept of "Duality Gap" can be used.

8.1 Duality

For a given minimization problem, which is referenced as the "primal" problem, a "dual" problem can be formed. The tools and the basics of Duality can be found in classical books on Convex Optimization [83, 84]. Usually a dual problem refers to the Lagrangian dual problem, which will be initially explained here to define the concepts of weak Duality, strong Duality and the Duality gap.

Let us consider the optimization problem in the standard form [83]:

$$\begin{aligned} &\text{minimize } f_0(x) \\ &\text{subject to } f_i(x) \le 0, \quad i = 1,...,m \\ &\qquad\qquad h_i(x) = 0, \quad i = 1,...,p \end{aligned} \tag{8.2}$$

with the variable $x \in \mathbb{R}^n$ where $f_0 \in \mathbb{R}^n \to \mathbb{R},\ f_i \in \mathbb{R}^n \to \mathbb{R},\ h_i \in \mathbb{R}^n \to \mathbb{R}$ are the objective, inequality constrains and equality constraints respectively.

The associated Lagrangian $(L_P : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R})$ cost function for this problem can be defined as follows:

$$L_P(x, \lambda, v) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} v_i h_i(x) \tag{8.3}$$

For this primal function, the dual function (Lagrangian dual function) can be defined as follows:

$$g(\lambda, v) = \inf_{x} \left[f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} v_i h_i(x) \right] \tag{8.4}$$

Therefore, for a given primal function $L_P(x, \lambda, v)$, the dual function can be defined as:

$$g(\lambda, v) = \inf_{x} L_P(x, \lambda, v) \tag{8.5}$$

A dual function is always concave because it is the point-wise infimum of a family of affine functions of (λ, v). If we define the optimal value of $f_0(x)$ to be $p*$, then the dual problem (dual problem provides a lower bound to the solution of the primal (minimization) problem) will always give lower bounds on that value. In other words, if x is primal feasible and (λ, v) is dual feasible, then,

$$f_0(x) - p* \leq f_0(x) - g(\lambda, v) \tag{8.6}$$

The solution of the dual problem gives a lower bound to the solution of the primal (minimization) problem. Strong Duality is defined for convex problems (optimizing

convex functions over convex sets) when both the primal and dual optimal values coincide. If we denote the optimal value of the dual function to be $d*$, then the property of strong Duality will hold when $p* = d*$. Weak Duality property will hold for the cases where $d* < p*$. The Duality gap (β) is defined to be the difference between primal and dual objectives associated with the primal feasible point x and dual feasible point (λ, v): $\beta = f_0(x) - g(\lambda, v)$. If the Duality gap is zero (when strong duality holds), then x is said to be primal optimal and (λ, v) is dual optimal.

Suppose an algorithm generates a set of primal feasible points $x^{(k)}$ and dual feasible points $(\lambda^{(k)}, v^{(k)})$ with $k = 1, 2, \ldots$. Since we know from equation (8.6) that the lower bound for $\beta^{(k)}$ is $f_0^{(k)}(x) - p*$, for any upper bound for $\beta^{(k)}$, $f_0^{(k)}(x) - p*$ will always be less than (or equal to) the upper bound. Therefore, if $\beta^{(k)} < \varepsilon, (\varepsilon > 0)$, $f_0^{(k)}(x) - p* < \varepsilon$ will also be satisfied, i.e., $f_0^{(k)}(x)$ will be close to $p*$ within a range of ε. This criterion can be used as the stopping condition where if $\beta^{(k)} < \varepsilon$, the algorithm can be terminated, and the final solution is said to be ε – optimal.

8.2 Legendre-Fenchel Transform

From a general perspective, a dual function can be recognized as a transformation/ mapping of the primal function to a different space. Among these transformations, we

use Legendre-Fenchel transforms to derive the dual function for our algorithm. This transform maps the space $(x, f(x))$ to the space $(y, f*(y))$, where $f*(y)$ is referred to as the Fenchel conjugate of $f(x)$. Under the assumption that this transform is reversible, one form can be regarded as the dual of the other.

For a given vector $x \in \mathbb{R}^n$, the Fenchel conjugate of $f(x)$ is defined as:

$$f*(y) = \sup_{x \in \mathbb{R}^n} \left\{ x^T y - f(x) \right\} \tag{8.7}$$

In order to find the dual function for our original cost function $L_{SGSL0}(x^l)$ (4.19) using the Fenchel conjugates, we use the following Fenchel-Rockafellar Duality Theorem [85].

Theorem 8.1 [85]

Let $f : \mathbb{R}^M \cup \{+\infty\} \to \mathbb{R}$ be a convex function and $g : \mathbb{R}^M \cup \{+\infty\} \to \mathbb{R}$ be a concave function (i.e. $-g$ is proper convex). Also, let $f*$ and $g*$ be the fenchel conjugates of f and g respectively. Then,

$$\inf_x \left\{ f(x) - g(x) \right\} = \sup_y \left\{ g*(y) - f*(y) \right\} \tag{8.8}$$

Using Theorem 8.1, the aforementioned Legendre-Fenchel Transformation function and the Karush-Kuhn-Tucker (KKT) conditions (KKT conditions are the first order necessary conditions for a solution in non-linear programming to be optimal), we hope to find the dual function for the primal problem stated in (4.19). This will enable us to find a suitable stopping criterion which can be used in our algorithm.

8.3 Dual Function of the SGSL0 Primal Function

Let us first re-write the overall cost function to be minimized for the SGSL0 model as follows:

$$\min_x L_{SGSL0}(x) = \min_x \begin{bmatrix} \frac{1}{2}\left\| y - \sum_{j=1}^{J} A^j x^j \right\|_2^2 + \lambda_1 \sum_{j=1}^{J} \left\| x^j \right\|_2 + \\ \lambda_2 \sum_{j=1}^{J} [n_j - \sum_{i=1}^{n_j} e^{\frac{-(x_i^j)^2}{2\sigma^2}}] \end{bmatrix} \tag{8.9}$$

As discussed in Chapter 4, we use the Block Coordinate Descent method to minimize the above objective function where each block is minimized iteratively while keeping the other blocks fixed. Due to the group optimization behavior, we find the duality gap for each block minimization at each iteration k and observe the behavior of the Duality gap $\beta^{(l,k)}$ where l is the group index. We can stop the iteration process if the Duality gaps for all the groups are below a certain threshold $\varepsilon > 0$.

Let us now consider the minimization for each group as found in (4.35) in Section 4.3.3 and refer to it as the Primal cost function - M^P:

$$\min_{x^{l,(k)}} M^P(x^{l,(k)}, x^{l,(k-1)}) = \min_{x^{l,(k)}} \begin{bmatrix} \frac{L_1}{2}\left\| x^{l,(k)} - x^{l,(k-1)} + \frac{\nabla d_1(x^{l,(k-1)})}{L_1} \right\|_2^2 \\ + \frac{L_2}{2}\left\| x^{l,(k)} - x^{l,(k-1)} + \frac{\nabla d_2(x^{l,(k-1)})}{L_2} \right\|_2^2 \\ + \lambda_1 \left\| x^{l,(k)} \right\|_2 \end{bmatrix} \tag{8.10}$$

Let us refer $\alpha_1 = x^{l,(k-1)} - \frac{\nabla d_1(x^{l,(k-1)})}{L_1}$, $\alpha_2 = x^{l,(k-1)} - \frac{\nabla d_2(x^{l,(k-1)})}{L_2}$ and re-write (8.10) for our convenience as

$$\min_{x^{l,(k)}} M^P(x^{l,(k)}, x^{l,(k-1)}) = \min_{x^{l,(k)}} \left[\frac{L_1}{2} \left\| x^{l,(k)} - \alpha_1 \right\|_2^2 + \frac{L_2}{2} \left\| x^{l,(k)} - \alpha_2 \right\|_2^2 + \lambda_1 \left\| x^{l,(k)} \right\|_2 \right] \quad (8.11)$$

We will now try to find the Dual function to M^P, which we would refer to as M^D. Let us denote $f(x^{l,(k)}) = \lambda_1 \left\| x^{l,(k)} \right\|_2$ to be the convex function and

$g(x^{l,(k)}) = -\left[\frac{L_1}{2} \left\| x^{l,(k)} - \alpha_1 \right\|_2^2 + \frac{L_2}{2} \left\| x^{l,(k)} - \alpha_2 \right\|_2^2 \right]$ to be the concave function. If $y^{l,(k)}$, $f^*(y^{l,(k)})$ and $g^*(y^{l,(k)})$ represent the dual variable, fenchel conjugates of f and g respectively, using Theorem 8.1 we can say,

$$\underbrace{\inf_{x^{l,(k)}} \left\{ f(x^{l,(k)}) - g(x^{l,(k)}) \right\}}_{primal\ problem} = \underbrace{\sup_{y^{l,(k)}} \left\{ g^*(y^{l,(k)}) - f^*(y^{l,(k)}) \right\}}_{dual\ problem} \quad (8.12)$$

Using (8.7),

$$\begin{aligned} g^*(y) &= \sup_x \left[y^T x - g(x) \right] \\ &= \sup_x \left[y^T x + \frac{L_1}{2} \left\| x - \alpha_1 \right\|_2^2 + \frac{L_2}{2} \left\| x - \alpha_2 \right\|_2^2 \right] \quad (8.13) \\ \frac{\partial g^*(y)}{\partial x} &= y + L_1(x - \alpha_1) + L_2(x - \alpha_2) = 0 \end{aligned}$$

The x which satisfies the KKT conditions as described in [20] is the solution of the above equation

$$x^* = \frac{L_1\alpha_1 + L_2\alpha_2 - y}{(L_1 + L_2)} \quad (8.14)$$

Substituting this x^* into (8.13), we can now write the dual for $g(x^{l,(k)})$ as

$$g^*(y^{l,(k)}) = \frac{1}{(L_1+L_2)}\begin{bmatrix} -\left\|y^{l,(k)}\right\|_2^2 + (y^{l,(k)})^T(L_1\alpha_1 + L_2\alpha_2) \\ + \frac{L_1}{2(L_1+L_2)}\left\|L_2(\alpha_2-\alpha_1) - y^{l,(k)}\right\|_2^2 \\ + \frac{L_2}{2(L_1+L_2)}\left\|L_1(\alpha_1-\alpha_2) - y^{l,(k)}\right\|_2^2 \end{bmatrix} \tag{8.15}$$

To find the dual for $f(x^{l,(k)})$ we use the dual norm property proved in [83]. It states that if $\|.\|_p$ is a norm on $\circ^n$, with dual norm $\|.\|_q$ where $\frac{1}{p}+\frac{1}{q}=1$, then the conjugate of $h(x)=\|x\|_p$ is

$$h^*(y) = \begin{cases} 0 & \|y\|_q \le 1 \\ \infty & o.w \end{cases} \tag{8.16}$$

Also, if a function $\bar{h}(x) = \lambda h(x),$ then its dual can be written as $\bar{h}^*(y) = \lambda h^*(y/\lambda).$

Therefore, the dual for $f(x^{l,(k)})$ can be written as

$$f^*(y^{l,(k)}) = \begin{cases} 0 & \left\|\frac{y^{l,(k)}}{\lambda_1}\right\|_q \le 1 \\ \infty & o.w \end{cases} \tag{8.17}$$

Now using (8.12) we can write the dual problem of maximizing M^D w.r.t $y^{l,(k)}$ as

$$\max_{y^{l,(k)}} M^D(y^{l,(k)}) = \max_{y^{l,(k)}} \left\{g^*(y^{l,(k)}) - f^*(y^{l,(k)})\right\} \tag{8.18}$$

Therefore, now we can observe the duality gap (β) between M^P and M^D to determine the iteration number k which gives the desired stopping criterion.

8.4 Simulation Studies

During the minimization of the primal cost function M^P, we compute and plot the corresponding dual cost function M^D for each group for every iteration. We can observe that when the iteration count k increases, the distance between the primal and dual cost functions gradually decrease.

For this experiment we used the SGSL0 algorithm with the sparsity level $k = 80$, the number of sensors $n = 380$, and the number of sources $m = 800$. We used 16 groups with each having a group length of $n_l = 50$ for simplicity and kept the regularization terms fixed. It is important to note that, by observing (8.17) and (8.18), when the condition $\left\| \frac{y^{l,(k)}}{\lambda_1} \right\|_q \leq 1$ is not satisfied for a given iteration k, the dual function would become $-\infty$. This behavior is discussed in [92] and [20]. During our experiments we too experienced this behavior where for certain iterations there would not be a dual feasible point for the primal. But after considering the overall set of iterations for all the groups, it was evident that the dual function exhibits concavity. For clarity, we plot both primal and dual together and the dual separately, as they show a substantial difference in the beginning. We also replace the $-\infty$ values in the dual function by 0's to preserve clarity in the plots. The following are the plots of the Primal and Dual functions against the number of iterations (k).

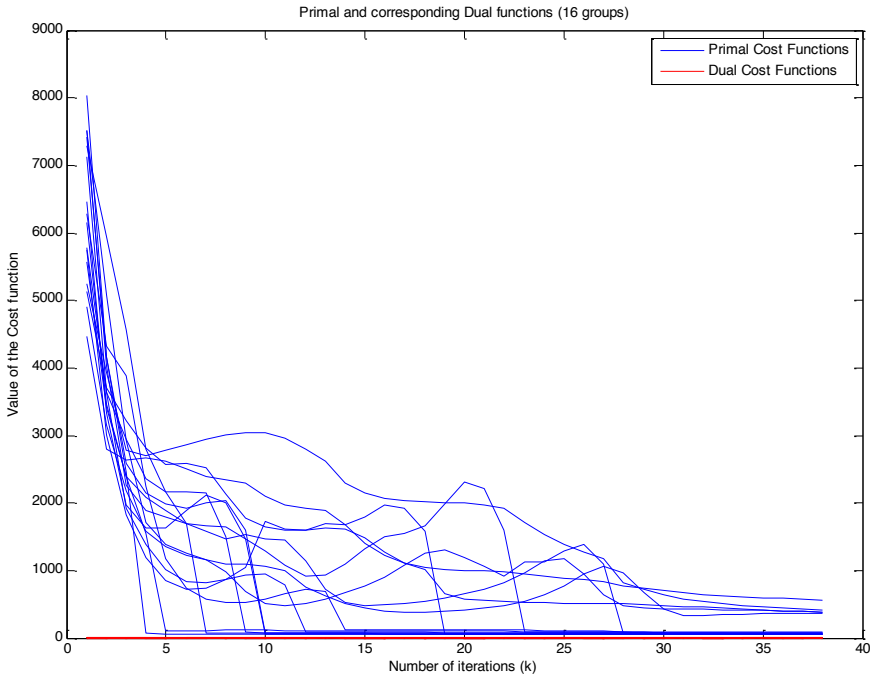


Figure 8.1: Primal and Dual cost functions for $k = 36$ iterations

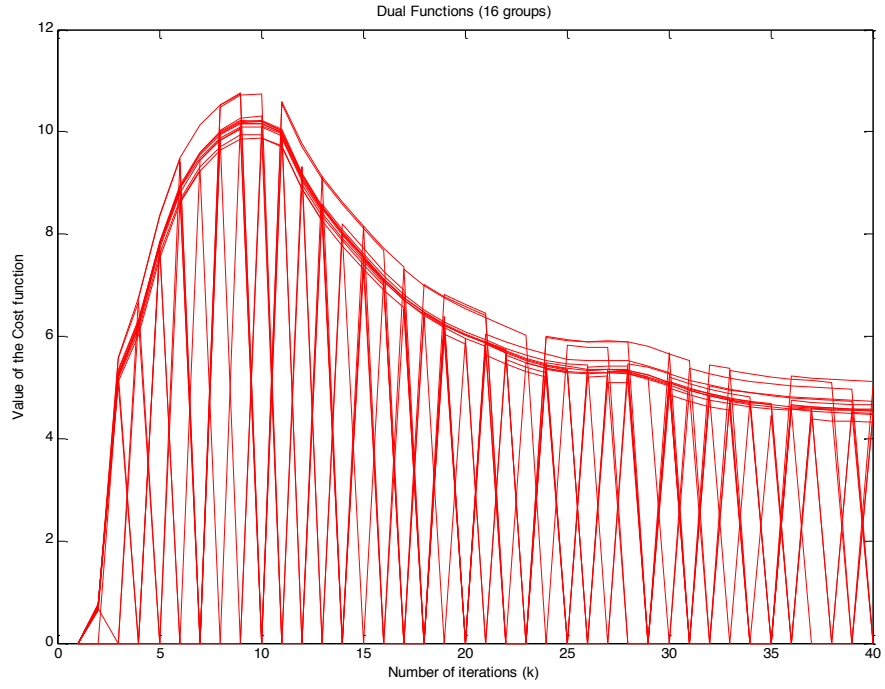


Figure 8.2: Dual Cost functions for $k = 36$ iterations

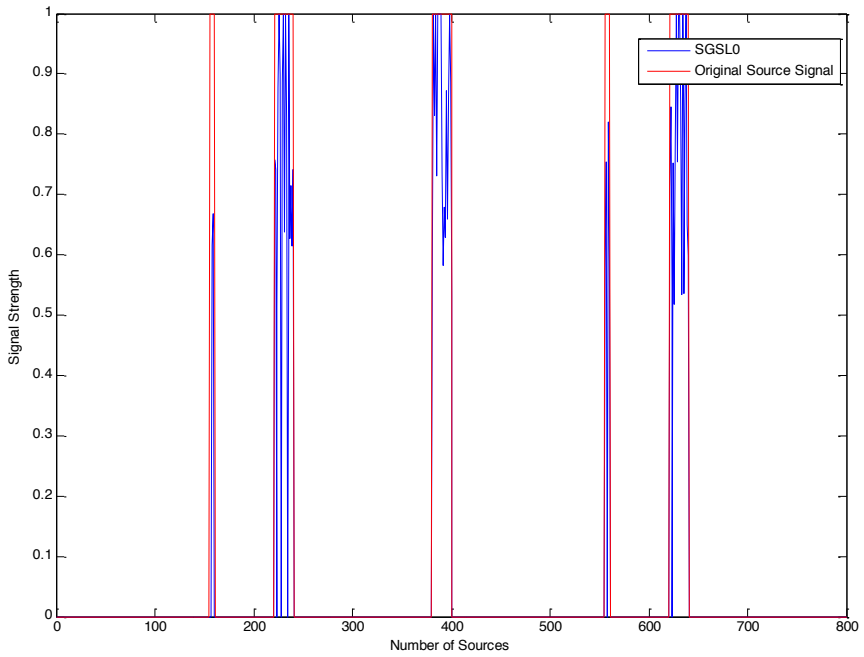


Figure 8.3: Reconstructed Signal stopped at $k = 36$ iterations

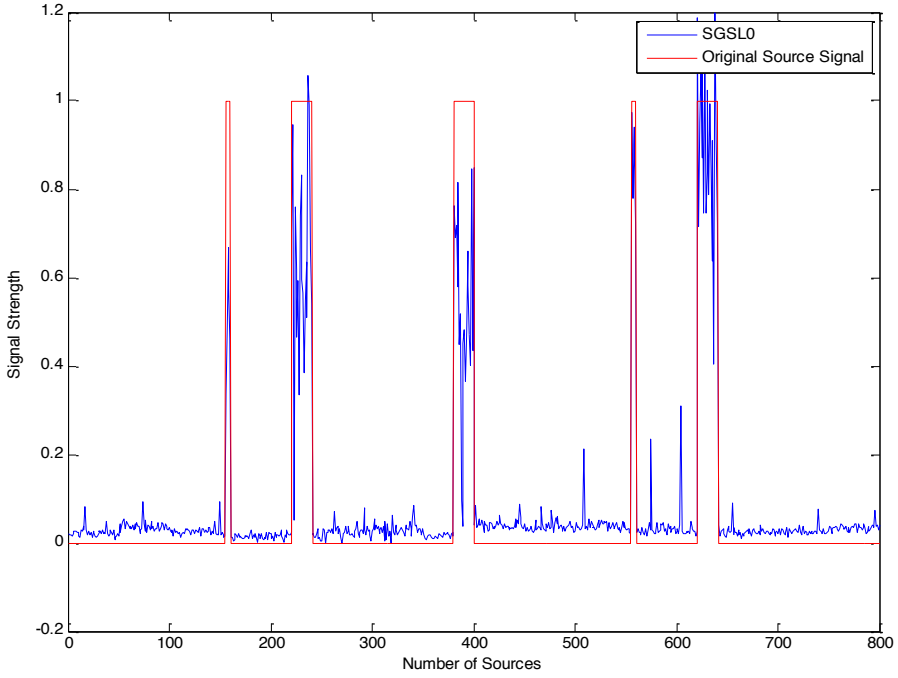


Figure 8.4: Reconstructed Signal stopped at $k = 18$ iterations

We can see from comparing Figures 8.1 and 8.4 that the reconstruction improves when we increase the number of iterations from 18 to 36. This is justified by observing the decrease in the duality gap from iterations 18 to 36. Therefore, we can use this criterion to help us determine when to stop the optimization process, especially when we do not have *a-priori* knowledge about the original signal.

It is also important to note that, we tried randomizing the sequence of the groups being iterated, hoping that the stopping criterion would reach faster in some instances than the other. We used the *randperm()* function in Matlab to randomly shuffle the 16 groups during this process. But we were not able to witness a significant improvement by randomizing the sequence of groups against iterating them from 1 to 16 sequentially. The multiple lines in Figures 8.2 and 8.3 represent these group minimizations.

9 CONCLUSION AND FUTURE WORK

9.1 Conclusion

We believe the work carried out in this thesis addresses the "sparse signal reconstruction using non-convex regularizers" problem in a holistic manner from a more theoretical perspective. Our initial attempt was to theoretically prove that the l_0 -norm (or l_0 -norm based) regularizers produce better reconstruction than the l_1-norm based regularizers with respect to the number of measurements needed. Once we had discussed the importance and advantages of using such non-convex regularizers, we attempt to tackle the challenging task of achieving global convergence in the optimization step.

Furthermore, we introduce a novel algorithm which reconstructs signals having both group-wise and within group sparsity behavior. The motivation for this was the structure of the MEG signals generated by the active brain sources. We term this algorithm as the Sparse Group Smoothed l_0 (SGSL0) algorithm, which is flexible enough to handle any level of group/ within group sparsity by changing its corresponding regularization parameters accordingly. An important novel contribution related to this

algorithm is the introduction of the global convergence criterion. This can be verified to avoid unnecessary iterations through-out the algorithm.

In Chapter 5, we were able to show that this novel algorithm performs better than the conventional l_1 -norm counterpart using a wide-range of simulations. As an extension to the source signal recovery for a given time point, we also show how to recover a non-stationary signal by stacking the source matrix into a super vector. Additionally, in Chapter 6 we discuss the theory related to the Smoothed l_0 -norm regularization and its global convergence. This enabled us to formalize a theoretical basis for a more comprehensive global convergence criterion for the SGSL0 algorithm.

Finally, we incorporate novel Information criteria techniques and concepts of Duality to find the best set of regularization parameters and a proper stopping criterion respectively, for a given signal reconstruction problem. In Chapter 7, we were able to successfully illustrate that the regularization parameters (models) with lower information criteria performs better than the ones with higher information criteria. We believe this will benefit profoundly when we have limited *a-priori* knowledge about the signal being reconstructed. Also, Chapter 8 provides the necessary tools to determine when to stop the algorithm, which is an important contribution considering the non-differentiability of the objective function.

9.2 Future Work:

As we have mentioned before, we assume the groups to be non-overlapping when devising our novel algorithm. As a future extension, SGSL0 can be modified to cater for the overlapping groups case as well. Pioneering work related to the overlapping groups case was carried out by Yuan et. al [93] and Jacob et. al [94], where we believe that similar constructions can be emulated for the SGSL0 algorithm as well.

In particular, Jacob et al. [94] modifies the group lasso [25] penalty, which we would briefly explain in this section.

Let us consider the vector to be reconstructed as $w \in \mathbb{R}^p$. Let us also define g to be a subset of the entries of w. In other words, a group g can take any subset from the power set $P([1, p])$. A power set $P(S)$ is defined to be the set of all subsets of S. We also define G to be a group of such subsets, usually given as *a-prior* information for a given problem. Two overlapping groups would have at least one coefficient in common. For $w \in \mathbb{R}^p$ and $g \subset G$, $w_g \in \mathbb{R}^p$ is defined as the vector whose entries are the same as w for the coefficients in g, and are 0's for the rest of the coefficients. Also, $V_G \in \mathbb{R}^{p \times G}$ is defined as the set of $|G|$ tuples of vectors $v = (v_g)_{g \in G}$ where each $v_g \in \mathbb{R}^p$ satisfies $\text{supp}(v_g) \subset \text{g}$ for each $\text{g} \in G$. Thereby, Jacob et al. [94] replaces the group lasso (non-overlapping) penalty (9.1) by the group lasso (overlapping) penalty (9.2) as follows:

Group Lasso (Non-overlapping) Penalty [25]:

$$\Omega_{group}^{G}(w) = \sum_{g \in G} \left\| w_g \right\|_2 \tag{9.1}$$

Group Lasso (overlapping) Penalty [94]:

$$\Omega_{overlap}^{G}(w) = \inf_{v \in V_G, \sum_{g \in G} v_g = w} \sum_{g \in G} \left\| v_g \right\|_2 \tag{9.2}$$

It can be seen that, when the groups do not overlap, $w = \sum_{g \in G} v_g$ with $\mathrm{supp}(v_g) \subset g$. In other words, $v_g = w_g$ for all $g \in G$ and (9.2) degenerates to (9.1).

Therefore, following the above modification, we can extend SGSL0 for the non-overlapping case as well.

Another future extension would be to incorporate Bayesian inference in the reconstruction model and to better estimate the initial approximation of the solution. Unprecedented work related to Bayesian related reconstruction modeling can be seen in [95-98]. Using these literatures, one can explore the possibility of pairing Bayesian inference with the SGSL0 algorithm as future work.

Finally, one could improve the convergence rate of the SGSL0 algorithm by replacing the initial approximation with a solution from an Orthogonal Matching Pursuit (OMP) based method like Group OMP [91]. Although these methods are less reliable, they inherit faster convergence rates.

BIBLIOGRAPHY

[1] Dale A. and Sereno M., "Improved localization of cortical activity by combining EEG and MEG with MRI cortical surface reconstruction: a linear approach" J. Cogn. Neurosci. 5 162–76, 1993.

[2] Scherg, M. And Von Cramon, D. "Evoked dipole source potentials of the human auditory cortex", Electroencephalogr. Clin. Neurophysiol. 65 344–360, 1986.

[3] Hämäläinen, M., Hari, R., Ilmoniemi, R. J., Knuutila, J. And Lounasmaa, O. V. "Magnetoencephalography theory, instrumentation, and applications to non-invasive studies of the working human brain", Rev. Modern Phys. 65 413–497, 1993.

[4] Jun, S. C., George, J. S., Pare´ -Blagoev, J., Plis, S. M., Ranken, D. M., Schmidt, D. M. and WOOD, C. C, "Spatio-temporal Bayesian inference dipole analysis for MEG neuroimaging data", NeuroImage 29 84–98, 2005.

[5] Mosher, J., P. Lewis, And R. Leahy, "Multiple dipole modeling and localization from spatio-temporal meg data", IEEE Transactions on Biomedical Engineering 39(6), 541–557, 1992.

[6] Vanveen, B., W. Van Drongelen, M. Yuchtman, And A. Suzuki, "Localization of brain electrical activity via linearly constrained minimum variance spatial filtering", IEEE Transactions on Biomedical Engineering 44, 867–880, 1997.

[7] Huang M, Aine Cj, Supek S, Best E, Ranken D, Flynn Er, “Multi-start downhill simplex method for spatio-temporal source localization in magnetoencephalography”, Electroencephalogr. Clin. Nuerophysiology 108(1):32-44, 1998.

[8] Wang J Z, Williamson S J And Kaufman L, “Magnetic source images determined by a lead-field analysis: the unique minimum-norm least-squares estimation”, IEEE Trans. Biomed. Eng. 39 665–75, 1992.

[9] Hämäläinen M And Ilmoniemi R, “Interpreting magnetic fields of the brain: minimum norm estimates”, Med. Biol. Eng. Comput. 32 35–42, 1994.

[10] Pascual-Marqui R D, Michel C M And Lehman D, “Low resolution electromagnetic tomography: a new method for localizing electrical activity of the brain”, Psychophysiology 18 49–65, 1994.

[11] Uutela K, Hämäläinen M And Somersalo E, “Visualization of magnetoencephalographic data using minimum current estimates”, Neuroimage 10 173–80, 1999.

[12] Gorodnitsky I F, George J S And Rao B D, “Neuromagnetic source imaging with FOCUSS: a recursive weighted minimum norm algorithm”, Electroencephalogr. Clin. Neurophysiol. 95 231–51, 1995.

[13] Tibshirani, R., “Regression shrinkage and selection via the lasso”, J. R. Statist. Soc. B, 58, 267–288, 1996.

[14] Bolstad, A., Veen, B. V. And Nowak, R., “Space-time event sparse penalization for magneto-/electroencephalography”, NeuroImage 46 1066–1081, 2009.

[15] Haufe S, Nikulin V V, Ziehe A, Müller K R And Nolte G, “Combining sparsity and rotational invariance in EEG/MEG source reconstruction”, Neuroimage 42 726–38, 2008.

[16] Ou W, Hämaläinen M And Golland P, “A distributed spatio-temporal EEG/MEG inverse solver”, Neuroimage 44932–46, 2009.

[17] Friston K, Harrison L, Daunizeau J, Kiebel S, Phillips C, Trujillo-Barreto N, Henson R, Flandin G And Mattout J, “Multiple sparse priors for the M/EEG inverse problem”, Neuroimage 39 1104–20, 2008.

[18] T. Auranen, A. Nummenmaa, M.S. Hamalainen, I.P. Jaaskelainen, J. Lampinen, A. Vehtari, M. Sams, “Bayesian analysis of the neuromagnetic inverse problem with l(p)-norm priors NeuroImage”, 26, pp. 870–884, 2005.

[19] Jeffs, B., R. Leahy, And M. Singh, “An evaluation of methods for neuromagnetic image reconstruction”, IEEE Transactions on Biomedical Engineering 34, 713–723, 1987.

[20] Gramfort, A., Kowalski, M., Hämäläinen, M., “Mixed-norm estimates for the M/EEG inverse problem using accelerated gradient methods”, Physics in Medicine and Biology 57, 1937e1961, 2012

[21] A. Gramfort, D. Strohmeier, J. Haueisen, M.S. Hämäläinen, M. Kowalski, “Time-frequency mixed-norm estimates: Sparse M/EEG imaging with non-stationary source activations”, NeuroImage, Volume 70, 15 April 2013, Pages 410-422, ISSN 1053-8119, http://dx.doi.org/10.1016/j.neuroimage.2012.12.051, 2012

[22] Wipf, D., Nagarajan, S.,” A unified Bayesian framework for MEG/EEG source imaging”, Neuroimage 44 (3), 947–966, 2009.

[23] J.M. Ales And A.M. Norcia, “Assessing direction-specific adaptation using the steady-state visual evoked potential: results from eeg source imaging”, Journal of Vision, 9(7)(8):1-13, 2009.

[24] B. R. Cottereau, J. M. Ales, And A. M. Norcia, “Increasing the accuracy of electromagnetic inverses using functional area source correlation constraints”, Hum. Brain Mapp, 33(11):2694{713, 2012.

[25] Ming Yuan, Yi Lin, “Model selection and estimation in regression with grouped variables”, Journal of the Royal Statistical Society: Series B (Statistical Methodology) Volume 68, Issue 1, pages 49–67, February 2006.

[26] Michael Lim, “The Group-Lasso: Two Novel Applications”, A Dissertation Submitted To The Department Of Statistics And The Committee On Graduate Studies Of Stanford University In Partial Fulfillment Of The Requirements for The Degree Of Doctor Of Philosophy, August 2013.

[27] J. Huang And T. Zhang, “The benefit of group sparsity”, Annals of Statistics, 38(4):1978-2004, 2006.

[28] Montoya-Martinez, J., Artes-Rodriguez, A., Hansen, L.K., Pontil, M., “Structured sparsity regularization approach to the EEG inverse problem”, In Proc. 3rd International Workshop on Cognitive Information Processing (CIP 2012), IEEE.

[29] E. Candes., “Compressive sampling”, In Proc. Int. Congress of Math., Madrid, Spain, Aug. 2006.

[30] E. Candes And J. Romberg, “Quantitative robust uncertainty principles and optimally sparse decompositions”, Found. Comput. Math., 6(2):227{254, 2006.

[31] E. Candes, J. Romberg, And T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information”, IEEE Trans. Inform. Theory, 52(2):489{509, 2006.

[32] E. Candes, J. Romberg, And T. Tao, “Stable signal recovery from incomplete and inaccurate measurements”, Comm. Pure Appl. Math., 59(8):1207{1223, 2006.

[33] D. Donoho, “Compressed sensing”, IEEE Trans. Inform. Theory, 52(4):1289{1306, 2006.

[34] Kutyniok G, “Theory and applications of compressed sensing”, GAMM Mitteilungen 36(1):79–101, 2013.

[35] Dmitry M. Malioutov, “A Sparse Signal Reconstruction Perspective for Source Localization with Sensor Arrays”, Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of Master of Science in Electrical Engineering and Computer Science at the Massachusetts Institute of Technology, July 2003.

[36] B. Grunbaum, “Convex polytopes”, Graduate Texts in Mathematics 221, Springer-Verlag, New York, 2003.

[37] D. L. Donoho And J. Tanner, “Neighborliness of Randomly-Projected Simplices in High Dimensions”, Proc. Natl. Acad. Sci. USA, 102:9452{9457, 2005.

[38] D. L. Donoho And J. Tanner, “Sparse Nonnegative Solutions of Underdetermined Linear Equations by Linear Programming”, Proc. Natl. Acad. Sci. USA, 102:9446{9451, 2005.

[39] D. L. Donoho And M. Elad, “Optimally sparse representation in general (nonorthogonal) dictionaries via l_1 minimization”, Proc. Natl. Acad. Sci. USA, 100:2197{2202, 2003.

[40] D. L. Donoho And X. Huo. “Uncertainty principles and ideal atomic decomposition”, IEEE Trans. Inform. Theory, 47:2845{2862, 2001.

[41] A. Cohen, W. Dahmen, And R. Devore, “Compressed sensing and best k-term approximation”, J. Am. Math. Soc., 22:211{231, 2009.

[42] E. J. Candes, “The restricted isometry property and its implications for compressed Sensing”, C. R. Acad. Sci. I, 346:589{592, 2008.

[43] S. Foucart, “A note on guaranteed sparse recovery via l1-minimization”, Appl. Comput. Harmon. Anal., 29:97{103, 2010.

[44] Candés, E.J., Wakin, M.B. And Boyd, S.P., “Enhancing sparsity by reweighted l1. Minimization Journal of Fourier Analysis and Applications”, v14. 877-905

[45] Rick Chartrand, “Fast algorithms for nonconvex compressive sensing: MRI reconstruction from very few data”, Proceedings of the Sixth IEEE international conference on Symposium on Biomedical Imaging: From Nano to Macro, p.262-265, June 28-July 01, 2009, Boston, Massachusetts, USA

[46] Hosein Mohimani , Massoud Babaie-Zadeh , Christian Jutten, “A fast approach for overcomplete sparse decomposition based on smoothed l_0 norm”, IEEE Transactions on Signal Processing, v.57 n.1, p.289-301, January 2009 [doi>10.1109/TSP.2008.2007606]

[47] H. Mohimani, M. Babaie-Zadeh, I. Gorodnitsky, C. Jutten, “Sparse recovery using smoothed l_0 (SL0): convergence analysis”, Preprint 2010.

[48] S. S. Chen, D. L. Donoho, And M. A. Saunders, “Atomic decomposition by basis pursuit,” SIAM J. Scientif. Comput., vol. 20, no. 1, pp. 33–61, 1999.

[49] E. Candes And J. Romberg, “l_1-Magic: Recovery of sparse signals via convex programming,” 2005, Online Available at: www.acm.caltech.edu/l1magic/downloads/l1magic.pdf

[50] Y. Li, A. Cichocki, And S. Amari, “Sparse component analysis for blind

source separation with less sensors than sources," in Proc. Int. Conf. Independent Component Analysis (ICA), 2003, pp. 89–94

[51] Laura B. Montefusco, Damiana Lazzaro, Serena Papi "A fast algorithm for non-convex approaches to sparse recovery problems", Journal Signal Processing, Volume 93 Issue 9, September, 2013, Pages 2636-2647

[52] S. Mallat And Z. Zhang, "Matching pursuits with time-frequency dictionaries," IEEE Trans. Signal Process., vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[53] S. Krstulovic And R. Gribonval, "MPTK: Matching pursuit made tractable," in Proc. Int. Conf. Acoustics, Speech, Signal Processing(ICASSP), Toulouse, France, May 2006, vol. 3, pp. 496–499.

[54] Sina Hamidi Ghalehjegh, Massoud Babaie-Zadeh, And Christian Jutten, "Fast block-sparse decomposition based on SL0", International Conference on Latent Variable Analysis and Signal Separation (ICA/LVA), pages 426–433, 2010.

[55] A. Eftekhari, M. Babaie-Zadeh, C. Jutten, And H. Abrishami Moghaddam, "Robust-SL0 for stable sparse representation in noisy settings," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., Apr. 2009, pp. 3433–3436.

[56] Andrew Blake And Andrew Zisserman, "Visual Reconstruction", The MIT Press Cambridge, Massachusetts, London, England, ISBN 0-262-02271-0 (1987), online:

[57] Brakken-Thal, S, "Gershgorin's theorem for estimating eigenvalues", http://buzzard.ups.edu/courses/2007spring/projects/brakkenthal-paper.pdf, 2007

[58] N. Simon, J. Friedman, T. Hastie, And R. Tibshirani, "A sparse-group lasso," Journal of Computational and Graphical Statistics, vol. 10,2012.

[59] L. Jacob, G. Obozinski, and J. Vert, “Group lasso with overlap and graph lasso”, In Proceedings of the 26th Annual International Conference on Machine Learning, pages 433–440. ACM, 2009.

[60] B. Sriperumbudur And G. Lanckriet, “On the convergence of the concave-convex procedure,” Neur. Inf. Process. Syst., 2009.

[61] http://en.wikipedia.org/wiki/Error_function

[62] http://mathworld.wolfram.com/Erf.html

[63] G. Gordon & R. Tibshirani, “Coordinate descent”, Available at: https://www.cs.cmu.edu/~ggordon/10725-F12/slides/25-coord-desc.pdf

[64] A. Beck And L. Tetruashvili, “On the convergence of block coordinate descent type methods”, Technical report. submitted to SIAM Journal on Optimization.

[65] A. Beck And M Teboulle, “Gradient-based algorithms with applications to signal recovery problems”, Yonina Eldar and Daniel Palomar, editors, Convex Optimization in Signal Processing and Communications. Cambridge University Press, 2010.

[66] Amir Beck , Marc Teboulle, “A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems”, SIAM Journal on Imaging Sciences, v.2 n.1, p.183-202, January 2009 [doi>10.1137/080716542]

[67] Y. E. Nesterov, “Gradient Methods for Minimizing Composite Objective Function”, CORE report, 2007

[68] http://dsp.ucsd.edu/~zhilin/MFOCUSS.m

[69] Oostenveld, R., Fries, P., Maris, E., And Schoffelen, J.M. (2011). “FieldTrip:Open source software for advanced analysis of MEG, EEG, and invasive

electrophysiological data", Comput. Intell. Neurosci. 2011, 156869

[70] Hämäläinen Ms, Sarvas J. "Realistic conductivity geometry model of the human head for interpretation of neuromagnetic data", IEEE Trans. Biomed. Eng. 1989;36:165–171.

[71] Ding L, He B., "Sparse source imaging in electroencephalography with accurate field modeling", Hum. Brain Mapp. 2008;29(9):1053–1067.

[72] http://webspace.ship.edu/cgboer/theneuron.html

[73] http://imaging.mrc-cbu.cam.ac.uk/meg/IntroEEGMEG

[74] J.M. Ales And A.M. Norcia, "Assessing direction-specific adaptation using the steady-state visual evoked potential: results from eeg source imaging", Journal of Vision, 9(7)(8):1-13, 2009.

[75] L. G. Appelbaum, A. R. Wade, V. Y. Vildavski, M. W. Pettet, And A. M. Norcia, "Cue-invariant networks for figure and background processing in human visual cortex", J Neurosci, 26(45):11695-708, 2006.

[76] T. Shimamura, H. Minami, And M. Mizuta, "Regularization Parameter Selection in the Group Lasso," [Online]. Available: http://www.stat.unipg.it/iasc/Proceedings/2006/COMPSTAT_Satellites/KNEMO/Lavori/Papers%20CD/Shimamura%20Minami%20Mizuta.pdf

[77] S. Negahban, P. Ravikumar, M. J. Wainwright, And B. Yu., "A unified framework for high dimensional analysis of M-estimators with decomposable regularizers", Advances in Neural Information Processing Systems, 2009.

[78] S. Negahban, P. Ravikumar, M. J. Wainwright, And B. Yu., “Supplementary material for a unified framework for high-dimensional analysis of M-estimators with decomposable regularizers”

[79] Soumyadeep Chatterjee, Karsten Steinhaeuser, Arindam Banerjee, Snigdhansu Chatterjee, Auroop Ganguly, “Sparse Group Lasso: Consistency and Climate Applications”

[80] I. Gorodnitsky And B. Rao., “Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm”, IEEE Trans. Signal Processing, vol. 45, pp. 600.616, March 1997.

[81] Sadanori Konishi, Genshiro Kitagawa, “Generalized information criteria in model selection”, Biometrika, 1996, 83, 4, pp. 875-890

[82] Kullback S., Leibler R. A.,“On information and sufficiency. Ann. Math. Statist. Assoc. 89, 550-9, 1951

[83] Boyd, S.,Vandenberghe, L., ”Convex Optimization”, Cambridge University Press, Cambridge 2004

[84] Dimitri P. Bertsekas., Non-Linear Programming: 2nd Edition, 1999, ISBN 1-886529-00-0

[85] Rockafellar Rt. “Convex Analysis”, Princeton University Press; 1972.

[86] G. Raskutti, M. J. Wainwright, And B. Yu., “Restricted eigenvalue conditions for correlated Gaussian designs”, Journal of Machine Learning Research, 11:2241–2259, August 2010.

[87] M. Hyder, K. Mahata, “An Improved Smoothed l_0 Approximation Algorithm for Sparse Representation” IEEE Transactions On Signal Processing, VOL. 58, NO. 4, April 2010.

[88] J. Liu, S. Ji, and J. Ye. SLEP: Sparse Learning with Efficient Projections. Arizona State University, 2009. http://www.public.asu.edu/jye02/Software/SLEP

[89] Richard Baraniuk, Mark Davenport, Ronald DeVore, Michael Wakin, "A Simple Proof of the Restricted Isometry Property for Random Matrices", 2006

[90] D Deedell, J.A. Tropp CoSaMP, "Iterative signal recovery from incomplete and inaccurate samples", 2008

[91] Aurelie C. Lozano, Grzegorz Swirszcz, Naoki Abe, "Group Orthogonal Matching Pursuit for Variable Selection and Prediction", 2009

[92] Malgorzata Bogdan, Ewout Van Den Berg, Weijie Su, Emmanuel J Candes, "Statistical Estimation and Testing via the sorted l_1 norm", 2013

[93] Yuan, L., Liu, J., and Ye, J., "Efficient methods for overlapping group lasso", In NIPS, 2011.

[94] L. Jacob, G. Obozinski, and J. Vert, "Group lasso with overlap and graph lasso", In International Conference on Machine Learning, 2009.

[95] Tipping, M. E., "Sparse Bayesian Learning and the Relevance Vector Machine", Journal of Machine Learning Research, 1:211–244, 2001.

[96] D. Wipf and B. Rao, "Sparse bayesian learning for basis selection," IEEE Trans. on Signal Processing, vol. 52, no. 8, pp. 2153–2164, 2004.

[97] Z. Zhang and B. D. Rao, "Exploiting correlation in sparse signal recovery problems: Multiple measurement vectors, block sparsity, and time-varying sparsity", In ICML 2011 Workshop on Structured Sparsity: Learning and Inference, 2011a.

[98] Z. Zhang and B. D. Rao, “Extension of sbl algorithms for the recovery of block sparse signals with intra-block correlation”, IEEE Transactions on Signal Processing, 2012.

APPENDIX

Using (6.6), (6.14) and (6.15), we show that (6.16) is true.

$$\gamma_A(n_0)+1 \le \max_{|\text{idx}|\le n_0} \frac{\sigma^2_{\max}(A)}{\sigma^2_{\min}(A_{idx})} = \frac{\|A\|_2^2}{\min_{|\text{idx}|\le n_0} \sigma^2_{\min}(A_{idx})} \tag{6.6}$$

$$P\left\{\sqrt{(m+n_l)/n_l}\,\sigma_{\max}(M) > 1+\sqrt{(m+n_l)/n_l}+\varepsilon\right\} \le e^{(-\frac{n_l\varepsilon^2}{2})} \tag{6.14}$$

$$P\left\{\sqrt{(m+n_l)/n_{0,l}}\,\min_{|idx|=n_{0,l}} \sigma_{\min}(M_{idx}) < 1-\sqrt{(m+n_l)/n_{0,l}}-r\right\} \le \binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})} \tag{6.15}$$

$$P\left\{\frac{n_{0,l}}{n_l}\gamma(n_{0,l}) > \frac{(1+\sqrt{(m+n_l)/n_l}+\varepsilon)^2}{(1-\sqrt{(m+n_l)/n_{0,l}}-r)^2}\right\} \le \binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})} + e^{(-\frac{n_l\varepsilon^2}{2})} \tag{6.16}$$

Let us define event A as: $\sigma^2_{\max}(M) > \dfrac{\left(1+\sqrt{(m+n_l)/n_l}+\varepsilon\right)^2}{(m+n_l)/n_l}$

and event B as : $\min_{|idx|=n_{0,l}} \sigma_{\min}{}^2(M_{idx}) < \dfrac{\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2}{(m+n_l)/n_{0,l}}$

Then we can say that,

$P(A) \le e^{(-\frac{n_l\varepsilon^2}{2})}$ and therefore, $P(\bar{A}) \ge 1-e^{(-\frac{n_l\varepsilon^2}{2})}$. Similarly,

$P(B) \le \binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})}$ and $P(\bar{B}) \ge 1-\binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})}$.

Event $\overline{B}$ can also be written as: $\frac{1}{\min_{|idx|=n_{0,l}} {\sigma_{\min}}^2(M_{idx})} \le \frac{(m+n_l)/n_{0,l}}{\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2}$

Using events $\overline{A}$ and $\overline{B}$ we can write the following:

$$P\left\{ {\sigma_{\max}}^2(M) \le \frac{\left(1+\sqrt{(m+n_l)/n_l}+\varepsilon\right)^2}{(m+n_l)/n_l} \right\} \ge 1-e^{(-\frac{n_l \varepsilon^2}{2})} \quad \text{(A.1)}$$

$$P\left\{ \frac{1}{\min_{|idx|=n_{0,l}} {\sigma_{\min}}^2(M_{idx})} \le \frac{(m+n_l)/n_{0,l}}{\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2} \right\} \ge 1-\binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l} r}{2})} \quad \text{(A.2)}$$

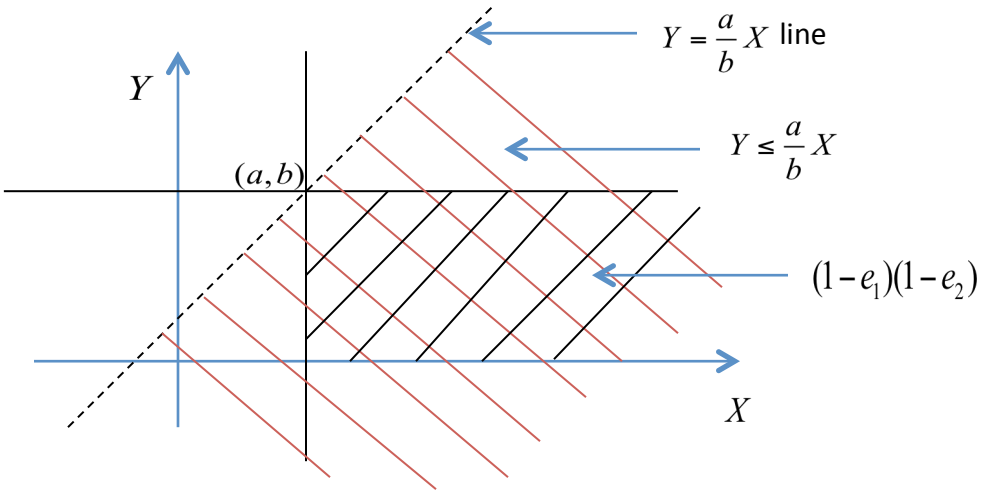


Figure A.1: Probability range description

We will now explain a probability relationship that will be used for subsequent analysis.

For independent events $Y \le a$ and $X \ge b$, let us say $P(Y \le a) \ge 1-e_1$ and

$P(\frac{1}{X} \le \frac{1}{b}) \ge 1-e_2 \Rightarrow P(X \ge b) \ge 1-e_2$. The probability range description for these two

events is illustrated in Figure A.1. Since the probability range for $(1-e_1)(1-e_2)$ is included in the range $Y \leq \frac{a}{b} X$, we can say the following holds true:

$P(\frac{Y}{X} \leq \frac{a}{b}) \geq (1-e_1)(1-e_2)$.

Therefore, using the same arguments on (A.1) and (A.2) we can say the following would hold:

$$P\left\{\frac{{\sigma_{\max}}^2(M)}{\min_{|idx|=n_{0,l}} {\sigma_{\min}}^2(M_{idx})} \leq \frac{n_l\left(1+\sqrt{(m+n_l)/n_l}+\varepsilon\right)^2}{n_{0,l}\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2}\right\} \geq \left(1-e^{(-\frac{n_l \varepsilon^2}{2})}\right)\left(1-\binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l} r^2}{2})}\right)$$

Therefore, from (6.6) we can say,

$$P\left\{\gamma_A(n_0) < \gamma_A(n_0)+1 \leq \frac{n_l\left(1+\sqrt{(m+n_l)/n_l}+\varepsilon\right)^2}{n_{0,l}\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2}\right\} \geq \left(1-e^{(-\frac{n_l \varepsilon^2}{2})}\right)\left(1-\binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l} r^2}{2})}\right)$$

$$P\left\{\gamma_A(n_0) > \frac{n_l\left(1+\sqrt{(m+n_l)/n_l}+\varepsilon\right)^2}{n_{0,l}\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2}\right\} < 1-\left(1-e^{(-\frac{n_l \varepsilon^2}{2})}\right)\left(1-\binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l} r^2}{2})}\right)$$

Therefore,

$$P\left\{\gamma_A(n_0) > \frac{n_l\left(1+\sqrt{(m+n_l)/n_l}+\varepsilon\right)^2}{n_{0,l}\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2}\right\} <$$

$$e^{(-\frac{n_l\varepsilon^2}{2})} + \binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})} - e^{(-\frac{n_l\varepsilon^2}{2})}\binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})}$$

Since $e^{(-\frac{n_l\varepsilon^2}{2})}\binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})} > 0$ the following inequality holds:

$$P\left\{\gamma_A(n_0) > \frac{n_l\left(1+\sqrt{(m+n_l)/n_l}+\varepsilon\right)^2}{n_{0,l}\left(1-\sqrt{(m+n_l)/n_{0,l}}-r\right)^2}\right\} < e^{(-\frac{n_l\varepsilon^2}{2})} + \binom{n_l}{n_{0,l}} e^{(-\frac{n_{0,l}r^2}{2})}.$$